

#### = 注意事項 =

国立研究開発法人 産業技術総合研究所(以下、産総研)は、本文書の記載内容に関して、一切の明示的または暗示的な責任または保証を負うものではありません。

本文書の一部または全部を、産総研の文書による承諾なく、無断で複写、複製、転載することを禁じ ます。本書の情報の使用に対して特許責任は一切負いません。また、本書は細心の注意を払って作成さ れていますが、産総研は、本書に掲載されている情報の正確性、誤り、記載漏れについての責任を負う ものではありません。また、本書に掲載されている情報は、将来予告なしに変更される場合があります。

MZ Platform をいかなる第三者に対しても、産総研の許可なく貸与、譲渡、その他の方法で使用させたり、担保に供したりしてはなりません。また産総研の許可なく、別のハードウェアに複製を作成することも禁じます。

本書中の社名、商品名などは、各社の登録商標または商標です。

1.	概要	要	.1
2.	デー	-タベース連携部の作成	.4
	2. 1.	「ローカル MYSQL 管理」複合コンポーネントの保存	.5
	2. 2.	「ローカル MySQL 管理」複合コンポーネントのロード	.7
	2. 3.	「ローカル MySQL 管理」複合コンポーネントの設定	11
	2. 4.	動作確認	15
	2. 5.	SQL 文実行機能の追加	21
	2. 6.	複数行一括追加機能の追加	28
	2. 7.	バックアップ機能の追加	33
	2. 8.	データベース内容設定機能の追加	40
	2. 9.	データベーステーブル編集機能の追加	17
3.	発泪	主依頼部の作成	57
	3. 1.	購買管理データベースの作成	57
	3. 2.	「発注依頼」複合コンポーネントの作成	59
	3. 3.	初期コンポーネントの追加	30
	3. 4.	公開メソッドおよびコンポーネント接続の初期設定	32
	3. 5.	トップ階層におけるイベント接続先の初期設定	36
	3. 6.	発注依頼フレームのメニュー登録	39
	3. 7.	参照機能の作成と登録機能作成の準備	72
	3. 8.	登録機能の作成	36
	3. 9.	削除機能の作成	92
	3. 10	. 更新機能の作成(登録機能の修正)	97
	3. 11	. 注文番号一覧設定機能の作成10	)0
	3. 12	. 金額計算機能の作成10	)6
	3. 13	. 伝票発行機能の作成1	11
4.	コン	/ポーネント接続関係(まとめ)1	25

#### 1. 概要

このチュートリアルでは、MySQL で作成した購買データベースと連携をとって購買伝票を発行するア プリケーションを作成します。アプリケーションの機能概要は以下の通りです。

なお、本書は MZ Platform の基本的な操作ができる方を対象に記載しております。コンポーネントの 接続方法や起動メソッドの設定方法等は「開発チュートリアル(基礎編)」または「アプリケーション ビルダー操作説明書」をご参照ください。また、MySQL の導入と設定につきましては、付属の「工程管 理システム導入マニュアル」をご参照ください。

### <u>メインウィンドウ</u>

MySQL データベースを操作するための機能を提供します。

- データベースへの接続
- データベースの新規作成
- テーブルの作成/更新/削除
- テーブルの一覧表示
- 選択されたテーブルの構成表示
- SQL 文実行機能
- データベーステーブルへの複数レコードー括追加機能
- データベースのバックアップ機能 ※ [データベース] メニューから実行
- データベーステーブル編集機能 ※ [データベース] メニューから実行

ータベース	購買	管理						_			×
elect * from	n irai				SQL文実	ίŦ				複数行	行追加
shonin	kibo	Kensa	seihinCD	kiboKoba	ii kiboKing	aku	kiboTanka	kiboSi	игуо	kiboN	ouki
	1 0		1		50,	000	10,000	1	5		
テーブル情 テーブル名 テーブル構	称	7	ィールド名(	英数字)	) (作成 ) デ <sup>ィ</sup> ・	 更	新創除	) +-		→覧設定	)
		1 173670	1 f78UR#		-•		21	リー覧		型変換→	<b>覧</b>
コーカルMy	/SQLIX	1 行追加 定	1 行首切除	)   デー:	-ト タベース作成	·=t	クエ (一・ 操作コマ	リ <b>→覧</b> 7ンド設	 	型変換→	<u>۴</u>
コーカルMy ドラ・	/SQL設 イバ [	1行追加 定 com.mysc	] 1 行韵原律 ql.cj.jdbc.Driv	ッ- デー: er	-ト タベース作成 ダンフ生成	·⊐t C:\P	クエ ビー・操作コマ rogram Files	リー覧 フンド設 WySQLV	定 MySQ	型変換→ L Server	覧 8.0\bi
コーカルMy ドラ・サ	/SQL設 イバ [ ーバ ]	1行追加 定 com.mysc dbc:mysc	1 行副除 ql.cj.jdbc.Driv ql://localhost/	er 3	-ト タベース作成 メンブ生成 IySaL起動	• ⊐t C:\P C:\P	クェ ニー・操作コマ rogram Files rogram Files	リー覧 7ンド設 WySQL\ WySQL\	定 MySQ MySQ	型変換→ L Server L Server	覧 8.0\bii
ローカルMy ドラ・ サ ユー・ パスワ	/SQL設 イバ ( ー-バ ) ザ名 「	1行追加 定 com.mysc dbc:mysc oot	1 行削除 ql.cj.jdbc.Driv ql://localhost/		-ト タベース作成 IVSQL起動 タベース作成	·⊐t C:\P C:\P crea	クェ ニー・操作コマ rogram Files rogram Files te database	リー覧 7ンド設 WySQL\ WySQL	定 MySQ MySQ sts	型変換→ L Server L Server	覧 8.0\bii 8.0\bii
ローカルMy ドラ・サ ユー パスワ データベー	/SQL設 イバ ( ゲーバ ) フード ( ス名 )	1 行追加 定 com.myse dbc:myse root	1 行助脉 ql.cj.jdbc.Driv ql://localhost/		-ト タベース作成 オンフ生成 IySOL起動 タベース作成 タベース指定	·⊐t C:\P C:\P crea use	クェ くー・操作コマ rogram Files rogram Files te database	リー覧 ?ンド設 WySQL1 WySQL1	定 定 MySQ sts	型変換→ L Server L Server	覧 8.0\bii

図 1 メインウィンドウ概観

### 発注依頼フレーム

発注依頼機能を提供します。メインウィンドウの [購買管理] - [発注依頼…] メニューを実行すると 発注依頼フレームが表示されます。

- irai テーブルの検索
- irai テーブルへのレコード登録/削除/更新
- 発注依頼の伝票発行機能

。) 発注依頼	<u>a</u>						
		参照	登録	削除 注注	文番号一覧設定	<b>宦 </b> [ (范	票発行…
注文番号	注文日	発注部門コード	購買品コード	発注数量	単価	金額	部門承認
A1	2005/12/20	S1	P1	5	10,000	50,000	0

図 2 発注依頼フレーム概観



図 3 伝票発行

項目	主キー	注文番号	注文日	発注部門コード	購買品コード	メーカ名
フィールド名	id	kobaiCD	iraibi	bumonCD	hinmokuCD	maker
データ型	Integer	String	Date	String	String	String

規格仕様	希望納期	発注数量	単価	金額
kikaku	kiboNouki	kiboSuryo	kiboTanka	kiboKingaku
String	Date	Integer	Double	Double

(仮)注文先コード	製品設備コード	検査の有無	部門承認
kiboKobaisakiCD	seihinCD	kiboKensa	shonin
String	String	Boolean	Boolean

図 4 irai(発注依頼)テーブル構成図

### 2. データベース連携部の作成

まずは、データベースとの連携を行うデータベース連携部を作成します。ここでは、付属の「工程管理システム簡易版」で使用されている「ローカル MySQL 管理」複合コンポーネントを流用することにします。この複合コンポーネントは、「データベースアクセス」コンポーネントや「外部プログラム通信」コンポーネントなどから構成され、MySQL データベースを操作するための機能を提供します。



① アプリケーションビルダーを起動します。

② ロードボタンをクリックし「AP\_DATA¥Production¥ProductionControl.mzax」をロードします。

ファイル 編集 アプリケーション・オブション・ヘルブ コメント行検索 アプリケーション名称 ■ アプリケーション	
アブリケーション名称	in an an an
₩ II<	
参照: ]] Production 🗸 🦸 🕫 🔚	
ProductionControl.mzax	
→ 一 一 一 た 値	
デスクトップ	
マイドキュ	
3-	
Tマイルタ: ProductionControl mzay     関(	
ネットワーク ファイルタイブ: オペエの総計研究に加速メ	
3 * COVERCE BERS / ** // (meas, meas, meas, meas, meas, meas, meas	
	<b> </b> ∪モート
	_

#### 図 5 ProductionControl.mzax のロード

### 2.1. 「ローカル MySQL 管理」複合コンポーネントの保存

「ローカル MySQL 管理」複合コンポーネントを流用するために、これをいったん別ファイルとして保存しておきます。

操作 「ローカル MySQL 管理」複合コンポーネントを保存しましょう。

- ① [階層選択] アイコンをクリックし、[アプリケーション階層選択] ダイアログを表示します。
- ② 「ローカル MySQL 管理」へ移動します。

WZ Platform アプリケーションビルダー - C:¥N	ZPlatform¥2.8¥AP_DATA¥Production¥ProductionControl.mzax	
ファイル 編集 アブリケーション オブション へ	レブ	コメント行検索
アプリケーション名称 工程管理簡易版		
アブリケーション       ドブリケーション       KEY: "工程管理防息版"	アブリケーション隔濁道沢         ● アブリケーション(KEY:"工程管理簡易版")         ● MySQL連携 [D.1] (KEY:"MySQL連携")         ● ローカルMySQL管理 [D.1-9] (KEY:"ローカルMySQL         ● 作業表 [D:7] (KEY:"工程表")         ● 目付書式変換 [D:7-49] (KEY:"目付書式変換")         ● 目付書式変換 [D:7-49] (KEY:"目付書式変換")         ● 目付書式変換 [D:7-30] (KEY:"日付書式変換")         ● 目付用食荷状況 [D:9] (KEY:"日付書式変換")         ● 日付別負荷状況 [D:0] (KEY:"日付指定")	・         ・
[アイコン凡例] 🛑 アブリケーション	  画面構成部品	
実行」実行(設定可)	画面編集   幌票編集   ロード   挿入   保存   上	書き保存 クリア 終了

図 6 「ローカル MySQL 管理」への移動

③ 複合コンポーネント「ローカル MySQL 管理」を保存します。

保存ボタンをクリックし、表示された [確認] ダイアログでこの階層のみを選択します。

ファイル名を「MySQL.mzcx」として保存します。拡張子「.mzcx」は、保存時に自動的に付加されます。

WZ Platform アプリケーションビルダー - C:¥MZPlatform¥2.8¥AP_DATA¥Production¥ProductionControl.mzax	
ファイル 編集 アブリケーション オブション ヘルブ	コメント行検索
コンボーネント名称 ローカルMySOL管理	
■ ローカル/MyGOL管理 ID: 1-9 KEY: "ローカル/MySOL管理"	
上位厚へ伝播するイベント 処理先Tイベント:データベースアクセスからの処理先Tイベント(イベント番号:0,1,-1) ダンブ生気があよびダンブ実行のエラー(イベント番号:-2,-3、処理先Tデータは指定されているダンブ データ生成イベント:データベースアクセスからのデータ生成イベント(イベント対象データが検索編集)	コマンドあるL v3MySoL実行コマンド)
データ設定イベント、タンブ生成時(イベント対象データがダンプ文字列) データ更新イベント:タンブ実行あよびデータペース構成更新時	₩ <b>保存</b>
	(%77: 🍌 Production 👻 🌶 💬 🚍
D:1+91  住Y:3連携券間野   マントを伝播され   オペて この階層のみ キャンセル	<b>最近使った項</b> 目
	7201-17
指定した文字列と連結して	マイドキュメント
指定した文字列と連結して	「「 コンピュー ター
	(戦) ファイル名: (MySOL) (保存
データ生成イベント イベントを伝播させ	ネットワーク ファイルタイブ: MZ復合コンボーネントXML(mzcx) ・ 取消し
【アイコン凡州】 ●アブリケーション ●画面構成部品 ●ウィンドラ ●メニュー ●バネル ●処3	
実行」実行(設定可) 画面編集 候票編集 ロード 神大 (保存	上書き保存 クリア 終了

図7複合コンポーネント「ローカル MySQL 管理」の保存

④ アプリケーションビルダーのトップ画面に移動し、
 クリアボタンをクリックして、
 画面をクリアします。

# 2.2. 「ローカル MySQL 管理」複合コンポーネントのロード

アプリケーションビルダートップ画面に「フレーム」コンポーネントを配置し、アプリケーション起 動時に「フレーム」が表示されるように接続します。

## 操作 コンポーネントを追加し、接続しましょう。

- 必要なコンポーネントを追加します。
   作業領域で右クリック-[コンポーネント追加] [画面構成部品] [ウィンドウ] [フレーム]
   とクリックします。
- ② 下の表に示すようにコンポーネントを接続します。

開始

接続項目	接続関係
接続元コンポーネント	
(イベント発生コンポーネント)	
発生イベント	アプリケーション開始イベント
接続先コンポーネント	■フレーム (ID:1)
起動メソッド	フレームを表示する()

終了

接続項目	接続関係
接続元コンポーネント	$\square$ $\square$ $\downarrow$ $\downarrow$ $\downarrow$ (ID:1)
(イベント発生コンポーネント)	
発生イベント	アクションイベント
接続先コンポーネント	■アプリケーション
起動メソッド	アプリケーションを終了する()

確認	接続関係と配置は以下のようになります。	
	MZ Platform アプリケーションビルダー	×
	ファイル 編集 アブリケーション オブション ヘルブ コメント行検索	-
	アブリケーション名称	Ľ.
	アプリケーション アプリケーション アプリケーション の後もイベント フレームを表示する コフレーム	
	KEY:""	JI
	アクションイベンド アブリケーションを終了する アブリケーション	.)
	(KEY:"21/-41"	ון נ
	【アイコン凡例】 🎒 アブリケーション 📄 画面構成部品 🛑 ウィンドウ 🧾 メニュー 🗐 パネル 🗐 処理部品 🌒 様合(画面構成) 🗐 様合(処理) 🛑 リモート	
	実行 実行 (設定可) 西面編集 「「県栗編集 ロード」 挿入 (保存 上書き(保存 クリア) 終了)	

図 8 フレームコンポーネントの配置

操作 「ローカル MySQL 管理」複合コンポーネントを追加し画面に配置しましょう。

- 前節でファイルに保存した「ローカル MySQL 管理」複合コンポーネント「MySQL. mzcx」を ロードします。アプリケーションビルダー画面上で 右クリック-「複合コンポーネント追加]-「ロード…」とクリックします。
- ② 表示された[開く]ダイアログから、「MySQL.mzcx」を選択し、開くボタンをクリックします。

👑 MZ Platform アプリケーションビ	レター							
ファイル 編集 アブリケーション オ	ブション ヘルブ			コメント行検索		•		
アゴリケーション名称					6			
■ アブリケーション	アプリケーション開始イイ	(V) F	フレームを表示する	27	レーム			
KEY:"	)			KEY	:"フレーム1"			
17V-4	アクションイベント		アブリケーションを終了する	7	ブリケーション			
KEY:"71-41"	J			KEY				
コノホーイノト 垣加 コンポーネントー 新潟tha		₩ 開<						X
複合コンボーネント作成		赤昭・ 🚺	Production					
複合コンボーネント追加	グラン・					- 18 <sup>-</sup>		
取り付け	チュートリアル ♪ テーブル ♪	● (使った)面	MySQL mzcs MySQL mzck					
すべて展開	合成モジュール ♪	B	<b>A</b>					
すべて閉じる	稼働日時 ▶							
コンボーネント検索	金型履歴	7271-17						
GUIコンポーネント検索	D-F							
コメント行追加								
コメント行検索		マイドキュ メント						
コンボーネントID再設定		1						
1		32Ka-						
		ター	$\sim$					
		77	イル名: MvSQL.mzcx					
		ネットワーク ファ	イルタイプ: オペイの時込ま	「能な雑会コンポーネ」	ットファイル (mzcx	mzcs xml cmn	-	
			[2 COMPCT		2 , 2 7 i je <sub>t</sub> unzor,			
[アイコン凡例] 🛑 アブリク	アーション 📄 画面構成部品 📄 🖞	インドウ 🥑メニュー	- 🗐パネル 🗐処理部品	■複合(画面構成)	2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	リモート		
実行	実行(設定可) 画面編集 帳	[編集] ロード	· 「 挿入 「 保存 」 上書:	き保存 クリア	\$ <b>8</b> 7			

図 9 「ローカル MySQL 管理」複合コンポーネントのロード

確	認	接続関係と配置は以下のようになります。

🖖 MZ Platform アプリケーションピルタ	y—			
ファイル 編集 アブリケーション オブ	ション ヘルブ		コメント行検索	
アブリケーション名称				
<b>アブリケーション</b>	アプリケーション開始イベント	フレームを表示する	<b>ノ</b> フレーム	
	70992442F	アプリケーションを終了する		
RET. JU-AT			(Ker.	)
ID:2 KEY:"ローカルMySQL管理"				
【アイコン凡例】 🛑 アブリケー	-ション 📄 画面構成部品 🛑 ウィンドウ 📒	▶メニュー	] 複合(画面構成) 🛛 複合(処理	) <mark>=</mark> Vモート
実行実行	行(設定可) 画面編集 帳票編集	ロード 挿入 保存 上書き(	R存 クリア 終了	

図 10 「ローカル MySQL 管理」複合コンポーネントの配置

③ 画面編集ボタンをクリックし、画面編集画面で「ローカル MySQL 管理」複合コンポーネントを フレームに追加します。

集						
<ul> <li>フレーム [ID:1] (KEY:"フレーム1")</li> <li>コーカルMSO(登録 ID:2) (KEY:"コーカルMs</li> </ul>	フレーム [ID:1] (KEY."フレーム1")					
Distribution of the state of	テーブル補報 テーブル名称 作成 更新 副版 テーブル構成 フィールド名(英数字) データ型 主キー	一覧設定				
	1行動物         1行動除         ソート         クェリー覧           ローカルMySQLI設定         データベース作成・コピー・操作コマンド設定           ドライパ         com.mysql.cj.jdbc.Driver         ダンブ生成         MySQLIMySQL Server 8.0bir           サーパ         jdbc.mysql.//localhost/         MySQLIMySQL Server 8.0bir	型変換一覧 Nmysqldump.exe 8.0\bin\mysql.exe				
	ユーザ名 root パスワード データベース作成 create database if not exists データベース指定 use MySOL終了 quit					

図 11 「ローカル MySQL 管理」複合コンポーネントのフレームへの追加

④ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

⑤ 実行もしくは実行(設定可)ボタンをクリックして、アプリケーションが正しく起動する ことを確認します<sup>1</sup>。確認後はアプリケーションを終了します。

テーブル名称			作成	更新	削除	一覧設定
テーブル構成「	フィールト	*名(英数5	ヂ) デ <sup>.</sup>	- 2型	<u>主</u> +~	
	1行追加 11	行削除	V-F	010 -	クエリー覧	型変換一覧
コーカル <mark>MySQ</mark> L	1行追加 1 設定		ンート データベース作成	・コピー・排	クエリー覧 操作コマンド設定	型変換一覧
コーカルMySQL ドライバ	1行追加 1 設定 com.mysql.cj.jdbc	行前除 c.Driver	ソート データベース作成 ダンプ生成	・コピー・排 WySQLW	クェリー覧 衆作コマンド設定 ySQL Server 8.0\b	型変換一覧 pin\mysqldump.ex
コーカルMySQL ドライバ サーバ	1行追加 1 設定 com.mysql.cj.jdbc jdbc:mysql://local	行削除 C.Driver Ihost/	<u>ソート</u> データベース作成 <u>タンプ生成</u> MySQL起動	・コピー・打 NySQLMy	クエリー覧 衆作コマンド設定 ySQL Server 8.0\b GQL\MySQL Serve	型変換→覧 pin\mysqldump.ex er 8.0\bin\mysql.ex
コーカルMySQL ドライバ サーバ ユーザ名	1 行追加 11 設定 zom.mysql.cj.jdbc jdbc:mysql://local root	行削除 c.Driver lhost/	ソート       データベース作成       ダンフ生成       MySQL起動       データベース作成	・コピー・# INYSQL\M TFiles\MyS	クエリー覧 衆作コマンド設定 ySQL Server 8.0\b SQL\MySQL Serve abase if not exists	型変換→覧 pin\mysqldump.ex rr 8.0\bin\mysql.ex s
コーカル MySQL ドライバ サーバ ユーザ名 バスワード	1 行追加 11 設定 zom.mysql.cj.jdbc jdbc:mysql://local root	行前期条 こ.Driver lhost/	<ul> <li>ソート</li> <li>データベース作成</li> <li>ダンフ生成</li> <li>MySaL起動</li> <li>データベース作成</li> <li>データベース非式</li> </ul>	・コピー・# #MySQLWy 1 FilesWyS create data	クエリー覧 衆作コマンド設定 ySQL Server 8.0\L XQL\MySQL Serve abase if not exists	型変換一覧 pin\mysqldump.ex er 8.0\bin\mysql.ex s
コーカルMySQL ドライバ サーバ ユーザ名 パスワード データベース名	1 行為動加 11 設定 zom.mysql.cj.jdbc jdbc:mysql://local root production	行背脚条 c.Driver lhost/	<ul> <li>ソート</li> <li>データベース作成</li> <li>ダンフ生成</li> <li>MySQL起動</li> <li>データベース作成</li> <li>データベース指定</li> </ul>	・コピー・排 :WySQLWy 1 FilesWyS create data use	クエリー覧 衆作コマンド設定 ySQL Server 8.0\b SQL\MySQL Serve abase if not exists	型変換→覧 bin\mysqldump.ex er 8.0\bin\mysql.ex s

図 12 アプリケーション起動画面



① 保存ボタンをクリックします。

② 保存先フォルダを「AP\_DATA」にし、ファイル名を「購買管理.mzax」として保存します。

<sup>&</sup>lt;sup>1</sup> 購買管理 01.mzax

# 2.3. 「ローカル MySQL 管理」複合コンポーネントの設定

#### 【1】 初期化処理と終了処理の設定

アプリケーション開始時に初期化処理を、終了時に終了処理を行うようにイベント接続を行います。

### 操作 アプリケーションの初期化処理と終了処理を設定しましょう。

下の表に示すようにコンポーネントを接続します。

開始処理

接続関係
アプリケーション開始イベント
■ローカル MySQL 管理(ID:2)
初期化処理()

終了処理

接続項目	接続関係
接続元コンポーネント	
(イベント発生コンポーネント)	
発生イベント	アプリケーション終了イベント
接続先コンポーネント	■ローカル MySQL 管理(ID:2)
起動メソッド	終了処理()

- ② アプリケーション開始イベントから接続されている起動メソッドの順番を入れ替えます。 「フレーム」コンポーネントを「ローカル MySQL 管理」複合コンポーネントの下に配置します。
  - 確認 接続関係と配置は以下のようになります。

■アブリケーション	アプリケーション開始イベント	初期化処理	□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□□
KEY:""			ID:2 KEY:"ローカルMySQL管理"
		フレームを表示する	
			ID:1  KEY:"フレーム1"
	アプリケーション終了イベント	終了処理	「ローカルMySQL管理
			ID:2 KEY:"ローカルMySQL管理"
<u> </u>	アクションイベント	アプリケーションを終了する	アプリケーション
ID:1 KEY:"フレーム1"	•		KEY:"
□ローカルMySQL管理			
ID:2 KEY:"ローカルMySQL管理"			

図 13 初期化処理と終了処理の設定

#### 【2】 MySQL ダンプ生成コマンドのパス設定

次に、MySQL の mysql dump コマンドのパスを設定します。使用する MySQL のバージョンに合わせて パス設定します。

操作 MySQL の mysqldump コマンドのパスを設定しましょう。

- ① 実行もしくは実行(設定可)ボタンをクリックして、アプリケーションを起動します。
- ② [ダンプ生成]ボタンをクリックします。

表示された[開く]ダイアログで、MySQLのダンプ生成コマンドのパスを設定します。 MySQLを標準インストールした場合、そのパスは以下のようになります。

MySQL 8.0.x の場合: C:¥Program Files¥MySQL¥MySQL Server 8.0¥bin¥mysqldump.exe

2			-	o x				
テーブル情報								
テーブル名称		ind E	804	-nar				
テーブル構成	フィールド名(英数字)	データ型	±+-					
			2 開<					×
			ファイルの場所の	bin 「「ビクチャ 」とデオ 」 ビデオ			✓ Ø ♥ □・ a -	
	1 (7)808	V-F	最近使った	Windows (C:)			1	
ローカルMySQLJ ドライバ	統定 デ com.mysql.cj.jdbc.Driver	タンプ生成・コピ ダンプ生成 ・MyS	デスクトップ	Program Files MySQL MySQL Server 8.0				
リーハ ユーザ名 バスワード	Idbc:mysql//localhost/	MySQLI2動 1 File		mysql_upgrade.exe mysqladmin.exe mysqlbinlog.exe	6.95 MB アブリケー 6.87 MB アブリケー 7.21 MB アブリケー	2022/12/16 9:23 2022/12/16 9:22 2022/12/16 9:22		
データベース名	production	ータベース指定 use	F==XDF	mysqlcheck.exe mysqld.exe	6.88 MB アラリケー 52.2 MB アブリケー	2022/12/16 9:22		
Min	データベース解説作成	MySQL&?7 quit	PC	mysqld_mulu.pl mysqldump.exe mysqldumpslov.pl mysqlhamess_stdx.dll	28.1KB PLファイル 6.94MB アブリケー 7.69KB PLファイル 41.1KB アブリケー	2022/12/16 16:21 2022/12/16 9:22 2022/12/16 16:21 2022/12/16 9:22		
			2 7	アイル名(11):				- III <
			ネットワーク フ	ァイルのタイプロ:			~	取消

図 14 MySQL ダンプ生成コマンドパスの設定

## ワンポイント - ダンプ -

MySQL のデータは mysqldump というツールを使用することで、テキストファイルに出力することがで きます。ファイルの中身は MySQL でそのまま実行できる SQL 文が記述されているため、作成したダンプ ファイルを利用することで、データベースのバックアップと復元が可能となります。

#### 【3】 MySQL 起動コマンドのパス設定

MySQL の起動コマンドのパスを設定します。使用する MySQL のバージョンに合わせてパスを設定します。

操作 MySQLの起動コマンドのパスを設定しましょう。

- ① [MySQL 起動]ボタンをクリックします。
- ② 表示された[開く]ダイアログで、MySQLの起動コマンドのパスを設定します。
   MySQLを標準インストールした場合、そのパスは以下のようになります。

MySQL 8.0.xの場合: C:¥Program Files¥MySQL¥MySQL Server 8.0¥bin¥mysql.exe

			-	o x					
テーブル情報									
テーブル名称		rius 👳	N NOX	-11222					
テーブル構成	フィールド名 (英数	(字) データ型	±+-						
			2 開<						×
			ファイルの場所()	): 🛅 bin				< 🖉 🕫 🛄 •	
			a	名前	サイズ ス	目の種類	更新日時	ブレビュー	
L.	1 (7)410 1 (7)808	y~r	最近使った_	Ibprotobuf-life.dll	584 KB 7 921 KB L	アブリケー JB ファイル	2022/12/16 16:41 2022/12/16 16:41		
ローカルMySQL	RE	データベース作成・コピ		libprotobuf-lite-debug.dll	1.32 MB	アブリケー_	2022/12/16 16:22		
ドライバ	com.mysql.cj.jdbc.Driver	ダンブ生成 /MyS		libssi-1_1-x64.dll	674 KB	アブリケー_	2022/11/30 16:03		
サーバ	jdbc:mysql:/localhost/	Musco 8245	デスクトップ	Iz4_decompress.exe	6.05 MB	アブリケー	2022/12/16 9:21		
ユーザ名	tool	iny datagent in the	-	my_print_defaults.exe	5.99 MB	マラリケー	2022/12/16 9:23	1	
1/20-1	1004	データー、 ATEBS Creat		myisam_hoump.exe	6.35 MB 7	アブリケー	2022/12/16 9:22		
7.09-r		データペーン指定しいの	ドキュメント	myisamlog.exe	6.20 MB	アプリケー_	2022/12/16 9.22		
データベース名	production	i i i use	V-CHLOREDGO	myisampack.exe	6.26 MB	アブリケー_	2022/12/16 9:22		
1922	データベース新聞作成	MySQL&?7 quit		mysql.exe	6.98 MB 7	アブリケー_	2022/12/16 9:22		
				mysql_config_editor.exe	6.00 MB 7	アブリケー_	2022/12/16 9:23		
			PC	mysql_migrate_keyring.exe	6.95 MB	アブリケー	2022/12/16 9:23		
				mysql_secure_installation.exe	5.85 MB	r905	2022/12/16 9:23		
			2	ファイル名(N): mysql.exe					間く
			ネットワーク	ファイルのタイプロ:					取消

図 15 MySQL 起動コマンドの設定

### 【5】 ユーザ名、パスワード、データベース名の設定

ユーザ名とパスワードを、ご自身の MySQL の環境に合わせて設定します。

## 操作 ユーザ名、パスワード、データベース名を設定しましょう。

- ① アプリケーションビルダーのトップ画面に移動します。
- ② 実行もしくは実行(設定可)ボタンをクリックして、アプリケーションを起動します。
- ③ ユーザ名とパスワードを、ご自身の MySQL の環境に合わせて設定します。
- ④ データベース名に、さしあたり「mysql」と入力します。

#### 2.4. 動作確認

ここで、アプリケーションを少し操作してみることにします。各ボタンの機能は下図の通りです<sup>2</sup>。



図 16 アプリケーションの動作

- 操作 アプリケーションの動作を確認しましょう。
- [接続]ボタンをクリックし、データベースに接続します。
   この場合、接続先は「mysql」データベースです。
- ② [一覧設定]ボタンをクリックします。
   右上に、「mysq1」データベースに含まれるテーブルの一覧が表示されます。
- ③ テーブル一覧の中から、任意のテーブル名を選択します。 そのテーブルのフィールド名、データ型、主キー設定の有無が左側のテーブルに表示されます。

<sup>&</sup>lt;sup>2</sup> 購買管理 02.mzax

このアプリケーションからは、データベースやテーブルの作成を行うこともできます。新規にデータ ベース「test」を作成し、そこへ実際に「machine」テーブルを作成してみます。

操作 「test」データベースを作成しましょう。

① データベース名に「test」と入力します。

② [データベース新規作成]ボタンをクリックします。

テーブル情報一							
テーブル名称				作成	更新	削除	一覧設定
テーブル構成	71-	-ルド名(英数	(字)	デ	- タ型	主キー	
	1 行追加	1 行削除				クエリ→覧	型変換一覧
」 コーカルMySQL	1行追加 設定	1 行首明余	 _データベー	ス作成	・コピー・1	クエリー覧… 操作コマンド設定	型変換→覧
ー コーカルMySQL ドライバ	1 行追加 設定 com.mysql.cj	1行削除 .jdbc.Driver	ソート データベー ダンプ語	ス作成	・コピー・I [:\MySQL\M	クエリー覧 操作コマンド設定 IySQL Server 8.0\b	〕 型変換→覧 pin\mysqldump.exe
ー コーカルMySQL ドライバ サーバ	1 行追加 設定 com.mysql.cj jdbc:mysql:///	1 行消服余 i.jdbc.Driver localhost/	ソート データベー ダンプ3 MySQL3	ス作成 主成 起動	・コピー・I  ·MySQL\M  1 Files\MyS	クエリー覧 操作コマンド設定 IySQL Server 8.0\b SQL\MySQL Serve	型変換→覧 pin\mysqldump.exv r 8.0\bin\mysql.exv
ー コーカルMySQL ドライバ サーバ ユーザ名	1 行追加 設定 com.mysql.cj jdbc:mysql://	1 行資10余 i.jdbc.Driver localhost/	ソート       データペー       ダンブ3       MySQL3       データペー	ス作成 主成 起動	・コピー・I MySQLM I FilesMyS	クエリー覧 操作コマンド設定 IySQL Server 8.0\b SQL\MySQL Serve	型変換一覧 pin\mysqldump.exv r 8.0\bin\mysql.exv
1ーカルMySQL ドライバ サーバ ユーザ名 バスワード	1 行法趋加 最定 com.mysql.cj jdbc:mysql:/// root	1行削除 i.jdbc.Driver localhost/	<ul> <li>ソート</li> <li>データベー</li> <li>ダンブき</li> <li>Mysaui</li> <li>データベー</li> </ul>	ス作成 主成 起動 ス作成	・コピー・I :WySQLW 1 FilesWyS create dat	クエリー覧 操作コマンド設定 IySQL Server 8.0\b SQL\MySQL Serve tabase if not exists	型変換一覧 pin\mysqldump.exe r 8.0\bin\mysql.exe s
コーカルMySQL ドライバ サーバ ユーザ名 パスワード データベー <b>ズ</b> 名	1 行注血加 設定 com.mysql.cj <sup>1</sup> jdbc:mysql:// root ●●●●● 1 test	1 行脚除 i.jdbc.Driver localhost/	<ul> <li>ソート</li> <li>データペー</li> <li>ダンフき</li> <li>MySQLE</li> <li>データペー</li> <li>データペー</li> </ul>	ス作成 <sup>主成</sup> 短動 ·ス作成 ·ス指定	・コピー・I MySQLM I FilesMyS create dat use	クエリー覧 操作コマンド設定 IVSQL Server 8.0\b SQL\MySQL Serve tabase if not exists	型変換一覧 pin\mysqldump.exr r 8.0\bin\mysql.exr s

図 17 接続先データベースの変更

③ [接続]ボタン、続いて[一覧設定]ボタンをクリックします。
 「test」データベースに含まれるテーブルの一覧が空であることがわかります。

ここでは、「test」データベースに以下のようなテーブルを作成することにします。

テーブル名: machine

フィールド名、データ型、主キー属性:

フィールド名	id	name	entryDay
データ型	整数(Integer)	文字列(String)	日付(Date)
主キー属性	主キー(true)	非主キー(false)	非主キー(false)

#### !!!注意事項!!!

「ローカル MySQL 管理」複合コンポーネントでは、主キーを設定できるフィールドは1つで、そのデ ータ型は整数(Integer、Long)に限定されます。また、属性 auto\_increment が設定されます。

操作

「machine」テーブルを作成しましょう。

① テーブル名称に「machine」と入力します。



図 18 テーブル名称の設定

② [1行追加]ボタンをクリックします。

表に行が追加されます。

2				1000	
テーブル情報					
J=JN-647	machine	作成	更新 創院		一覧設定
テーブル構成	フィールド名(共動	<del>3) 7</del>	<u> 大型 </u> 主キ		
	4	java.lan	g.String		
<	1行政制 1行销修条	<u>v-</u> F	0I	リー覧	型変換一覧
п-ллмуsqu	1行追加 1行追加 皇贤定	ソート	クエ	リー覧 ?ンド設定	型实换一覧
ローカルMySQL ドライバ	1行追加 1行時修注 皇贤定 com.mysql.cj.jdbc.Driver	ソート データベース作成 ダンプ生成	クエ ・コピー・操作コマ :MySQLWySQLS	リー覧 7ンド設定 - erver 8.0\bi	型安換→覧
ローカルMySQL ドライバ サーバ	1行追加 1行始後 算定 com.mysql.cj.jdbc.Driver jdbc:mysql/ilocalhost/	ソート       データペース作成       ダンブ生成       MySQL超動	クエ ・コピー・操作コマ :MySQLWySQL S 1 FilesWySQLWyS	リー覧 ?ンド設定 erver 8.0\bi SQL Server	型変換一覧 in\mysqldump.e) 8.0\bin\mysql.e)
ローカルMySQL ドライバ サーバ ユーザ名	1171400 11734004 1872 20m.mysql.ci.jdbc.Driver 1.jdbc.mysql//localhost/ root	ソート           データペース作成           ダンフ生成           MySQL超数           データペース作成	クエ ・コピー・操作コマ :MySQLWySQL S 1 FilesWySQLWyS	リー覧 マンド設定 erver 8.0\bi SQL Server	型変換一覧 in\mysqldump.e) 8.0\bin\mysql.e)
ローカルMySQL ドライバ サーバ ユーザ名 バスワード	1175800 11758000 11758000 11758000 11758000 11758000 11758000 11758	ソート       データベース作成       ダン7生成       MySoL起か       データベース作成	クエ ・コピー・操作コマ ·MySQLMySQLS 1 FilesWySQLWyS create database i	リー覧 ?ンド設定 erver 8.0\bi SQL Server f not exists	型宏拗一覧 in\mysqldump.e) 8.0\bin\mysql.ex
ローカルMySQL ドライバ サーバ ユーザ名 バスワード データベース名	11万止20 11万止20 11万世紀 12万元 1月日の一 1月日の日 1月日の 1月日の 1月日の日 1月日の日 1月日の日 1月日日	ソート       データベース作成       ダンフ生成       MysolJBか       データベース作成       データベース指定	クェ ・コピー・操作コマ MySQLMySQL S 1 FilesMySQLMyS create database i use	リー覧 ?ンド設定 erver 8.0\bi SQL Server f not exists	型宏确一覧 in/mysqldump.ex 8.0\bin\mysql.ex

図 19 行の追加

③ 最初のフィールドを設定します。フィールド名「id」を入力します。データ型「java. lang. Integer」 をプルダウンリストから選択します。主キー属性は「true」のため、チェックボックスにチェック を入れます。

2.								×
テーブル情報								
テーブル名称	machine			作成	更新	資源余	一覧設定	:
テーブル構成	1 24	* ド名(英数	字)	デー	夕型	ŦŦ		
( ( )	0 id			java.lang.ln	teger			
				iava long S	tring	$\sim$		
				java.lang.ln	teger			
				Java.long.	nig (			
				Java.lang.Fi	oublo			
				java.rang.c	lioDecimal			
6		2	-	java.util.Da	te			200
	1行追加	1 行御除	- v-	java.lang.B	oolean	クエリー覧	型変換一	覧
ローカルMySQL	設定		データ	ベース作成	・コピー・排	キュマンド設	tie	
ドライバ	com.mysql.	cj.jdbc.Driver	1	ンプ生成	MySQL\My	SQL Server 8	.0\bin\mysqldu	mp.ex
サーバ	jdbc:mysql:	//localhost/	My	SQLIEB	1 Files MyS	QL\MySQL Se	rver 8.0\bin\m	/sql.ex
ユーザ名	root		データ	ベース作成	create data	baco if not or	inte	
バスワード					create data	base in not ex	1919	
データベース名	test		7-2	ヘー人指定	use			
10120	5-00-	フ新規作成		MySQL終了	quit			

図 20 フィールドの設定(id)

④ 2つ目のフィールドを設定します。[1行追加]ボタンをクリックします。
 先頭に行が新たに追加されます。

フィールド名は「name」、データ型は「java. lang. String」、主キー属性は「false」とします。

<b>5</b> .0					-		>
テーブル情報							
テーブル名称	machine		flust	更新	aubs.	一覧設定	
テーブル構成	name	/1-名 (英敔	字) · · · · · · · · · · · · · · · · · · ·	5판 a String	±+		
-	1 10		java.iarr	ginteger			
	1 1756,00	1 行首即余			クエリ→覧	型変換一覧	
」 コーカルMySQLI	1行追加 ( 設定	1 行背間余	 「データベース作成	・コピー・操	クエリー覧 作コマンド設定	型変換一覧	
コーカルMySQL」 ドライバ	1 行追加 設定 com.mysql.cj.	1行前即注 jdbc.Driver	ソート データベース作成 ダンプ生成	・コピー・操 -MySQL\MyS	クエリー覧 作コマンド設定 SQL Server 8.0\t	型変換一覧 pin\mysqldump.	.e)
」 コーカルMySQLI ドライバ サーバ	1 行追加 設定 com.mysql.cj. jdbc:mysql.//k	1 行首切余 jdbc.Driver ocalhost/	ソート データベース作成 ダンプ生成 MySQL短動	・コピー・操 :WySQLWyS 1 FilesWySG	クエリー覧 作コマンド設定 GQL Server 8.012 DLIMySQL Serve	型変換一覧 pin\mysqldump. er 8.0\bin\mysql	.e) .e)
コーカルMySQL」 ドライバ サーバ ユーザ名	1 行追加 設定 com.mysql.cj. jdbc:mysql://d root	1 行動瞭余 jdbc.Driver ocalhost/	ソート       データベース作成       ダンブ生成       MySQL起動       データベーフ作成	・コピー・揀 :WySQLWyS 1 FilesWySC	クエリー覧 作コマンド設定 GQL Server 8.0\b QL\MySQL Serve	型変換一覧 pin\mysqldump. er 8.0\bin\mysql.	.e) .e)
コーカルMySQL ドライバ サーバ ユーザ名 バスワード	1 行追加 於定 com.mysql.cj. jdbc:mysql://d root	1行御録 jdbc.Driver ocalhost/	ソート       データペース作成       ダンブ生成       MySOL超動       データペース作成	・コピー・操 :MySQLWyS 1 FilesWySC create datat	クエリー覧 作コマンド設定 GQL Server 8.01b DLIMySQL Serve pase if not exists	型変換一覧 pin\mysqldump. er 8.0\bin\mysql. s	.e) .e)
コーカルMySQL」 ドライバ サーバ ユーザ名 バスワード データベース名	1 行追加 設定 com.mysql.cj. jdbc:mysql:///d root ●●●● test	1 行首卿条 jdbc.Driver ocalhost/	ソート       データベース作成       ダンフ生成       MySQL超数       データベース作成       データベース指定	・コピー・操 :MySQLWyS 1 FilesWySC create datat use	クエリー覧 作コマンド設定 GOL Server 8.01b DLIMySOL Serve	型皮夠一覧 bin\mysqldump. or 8.0\bin\mysql. s	.e)

図 21 フィールドの設定(name)

⑤ 残りの1つのフィールドを設定します。[1行追加]ボタンをクリックします。
 <u>先頭に</u>行が新たに追加されます。
 フィールド名は「entryDay」、データ型は「java.util.Date」、主キー属性は「false」とします。



図 22 フィールドの設定(entryDay)

⑥ [作成]ボタンをクリックします。テーブルが作成され、一覧の最後尾にその名前が表示されます。

テーブル情報					_			
テーブル名称	ma	chine		(f#A	t 更新	副錄		一覧設定
テーブル構成 🔤		フィー	ルド名(英数	字)	データ型	主キー		machine
	0	entryDay		ja	ava.util.Date		1	
	1	name		jav	/a.lang.String			
	2	id		jav	a.lang.Integer	<ul> <li>Image: A set of the set of the</li></ul>		
		1751630	1 17809	V-F		クエリー覧	<b>.</b>	型变换一覧
コーカルMySQ	[]分]	1738AD	1 178002	<u> ソート</u> データベース	作成・コピー・	クエリー覧 操作コマント	 *設定一	型変換一覧
コーカルMySQ ドライル	LIQI	l fīš <u>ējo</u> Ē om.mysql.c	1 行首明余 j.jdbc.Driver	ソート データベース ダンプ生&	作成・コピー・ ま WySQLW	クエリー覧 操作コマント IySQL Serve	t "設定 r 8.0\bir	型変換一覧 n\mysqldump.i
コーカルMySQ ドライバ サーバ	LIBS S DO S Jo	l 173£10 E om.mysql.c Ibc:mysql:/	1 行首切除 j.jdbc.Driver /localhost/	ソート       データペース       ダンフ生系       MySQL設計	作成・コピー・ 和 MySQLW か FilesMy	クエリー覧 操作コマント NySQL Serve SQLMySQL	t * 設定 r 8.0\bir Server :	型変換一覧 h\mysqldump.i 8.0\bin\mysql.
コーカルMySQ ドライハ サーバ ユーザ名	LI設 に設 く jo	l (73£10 E om.mysql.c lbc:mysql:/ bot	1 行首明余 j.jdbc.Driver /localhost/	<ul> <li>ソート</li> <li>データペース</li> <li>ダンフ生成</li> <li>MySQL記録</li> <li>データペース</li> </ul>	作成・コピー・ MySQLW か FilesWy 作成 create da	クエリー覧 操作コマント IySQL Serve SQLWySQL tabase if not	ž <設定 r 8.0\bir Server : exists	型変換一覧 h\mysqldump.r 8.0\bin\mysql.r
ローカルMySQ ドライバ サーノ ユーザ名 パスワート	ن لللائ آ ال آ ال آ ال	I 173£10 E om.mysql.c Ibc:mysql:/ bot	1 行首切涂 j.jdbc.Driver /localhost/	ソート           データペース           ダンブ生床           MySaL記録           データペース	作成・コピー・ ま WySOLW け 1 FilesWy 作取な create da	クエリー覧 操作コマント NySQL Serve SQLWySQL tabase if not	፤ ≤設定 r 8.0\bir Server i exists	型変換一覧 n\mysqldump. 8.0\bin\mysql.
ローカルMySQ ドライバ サーバ ユーザ名 パスワート データベース名	i I I I I I I I I I I I I I I I I I I I	I fīt£h0 E om.mysql.c Ibc:mysql./ bot Dot	1 行首明余 j.jdbc.Driver /localhost/	ソート       データペース       ダンブ生成       MySQL規則       データペース	作成・コピー・ 類 MySQLW か 「FilesWy 作成 create da 指定 use	クエリー覧 操作コマント NySQL Serve SQLWIySQL tabase if not	¥ *設定 r 8.0\bir Server : exists	型変換一覧 n\mysqldump. 8.0\bin\mysql.

図 23 テーブルの作成

MySQL Command Line Client を起動し、実際にテーブルが作成されていることを確認します。

操作 テーブルが作成されていることを確認しましょう。

① MySQL Command Line Client を起動します。

[スタート] - [すべてのプログラム] または [すべてのアプリ] - [MySQL] - [MySQL 8.0 Command Line Client] とクリックします。

② パスワードを入力後、次のようにコマンドを実行して確認します。

mysql> use test;

mysql> show tables;

mysql> show columns from machine;

🔜 MySQL Com	mand Line Clie	ent				_ 🗆 ×
Your MySQL	connect i on	id is 1	0 to s	erver vers	ion: 4.1.15-nt	A
Type 'help;	'or '¥h' f	or help	. Туре	'¥c' to c	lear the buffer.	10
mysql> use Database ch mysql> show	test; anged tables;					
Tables_in	_test					
+   machine						
1 row in se	t (0.00 sec					
mysql> show	columns fr	om mach	ine;			
Field	+   Туре	Null	Key	+ Default	Extra	Ť.
entryDay   name   id	datetime   text   int(11)	YES   YES 	   PRI	NULL   NULL   NULL	auto_increment	-
3 rows in s	et (0.00 se	+ c)		t	+	÷t
mvsal>						

図 24 MySQL Command Line Client によるテーブル作成の確認

- 3 確認後、MySQL Command Line Clientを終了します。次のようにコマンドを実行します。 mysql> quit;
- アプリケーションを終了します。

## 2.5. SQL 文実行機能の追加

ここからは、「ローカル MySQL 管理」複合コンポーネントのメソッドを利用して、様々な機能を追加し ていくことにします。「ローカル MySQL 管理」複合コンポーネントの主な公開メソッドおよび発生する イベントは以下の通りです。

### <u>公開メソッド</u>

メソッド	説明
初期化処理()	この複合コンポーネントを使用するための準
	備を行います。使用前に必ず起動してくださ
	Ma.
終了処理()	この複合コンポーネントの使用を終えるため
	に必要な処理を行います。 <u>アプリケーション</u>
	終了時に必ず起動してください。
ローカル MySQL の接続・切断(int)	MySQL データベースシステムへの接続と切断を
	行います。引数に0を与えると切断、1を与え
	ると接続になります。
SQL 文を実行する(String)	与えられた SQL 文を実行します。検索などのよ
	うに、処理結果が返ってくる SQL 文を実行した
	場合、SQL 文実行結果は、この複合コンポーネ
	ントが発生するデータ生成イベントの内包デ
	ータとして取得できます。
イベント番号を指定して SQL 文を実行する	与えられた SQL 文を実行します。指定した整数
(String, int)	は、SQL 文実行結果が含まれるデータ生成イベ
	ントのイベント番号になります。
指定したテーブルに複数行を一括追加する	String で指定したテーブルに、PFObjectTable
(String, PFObjectTable)	に記述された複数の行(レコード)を追加しま
	す。
テーブルに記述した SQL 文を実行する	PF0bjectTable に記述された複数の SQL 文を順
(PFObjectTable)	次実行します。
ダンプ生成()	接続先データベースのダンプを生成します。
	生成されたダンプは、この複合コンポーネント
	が発生するデータ設定イベントの内包データ
	として取得できます。
ダンプ実行(Object,int)	object に与えたダンプ(データベースを作成す
	る一群の SQL 文)を実行し、データベースの内
	容を更新します。 <u>int は使われていません。</u>

<u>発生するイベント</u>

処理完了イベント

発生のタイミング	対象データ	イベント番号
データベースへの接続および切断	0	0
SQL 文エラーの発生	0	-1
実行結果を戻さない SQL 文の実行	更新された行数	0もしくは指定した番号

データ生成イベント

発生のタイミング	対象データ	イベント番号
実行結果を戻す SQL 文の実行	SQL 文実行結果	0もしくは指定した番号

データ設定イベント

発生のタイミング	対象データ	イベント番号
ダンプ生成()の実行	データベースのダンプ	0

データ更新イベント

発生のタイミング	対象データ	更新データ	イベント番号
ダンプ実行(Object, int)の実行	MySQL コマンドのフ	更新したデータ	0
によるデータベース内容の更新	ルパス名	ベース名	0

まずは、SQL 文の実行機能を追加します。<sup>3</sup>

#### 追加するコンポーネント

- テキストフィールド
- ボタン
- テーブル

#### 追加する機能

- 1. ボタンをクリックしたら、
- 2. テキストフィールドに記入されている SQL 文をイベント番号1を指定して実行し、
- 3. 実行結果をテーブルに設定する。

### 操作 コンポーネントを追加し、接続しましょう。

① 必要なコンポーネントを追加します。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [テキスト] [テキストフィールド] とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [ボタン] - [ボタン]
 とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [テーブル] - [テーブル]
 とクリックします。

 ② 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。
 [テキストフィールド]コンポーネントの[Columns]を「25」、[ComponentKey]を「SQL文」と
 設定します。

[ボタン] コンポーネントの[text]を「SQL 文実行」と設定します。

<sup>&</sup>lt;sup>3</sup> 購買管理 03.mzax

③ 下の表に示すようにコンポーネントを接続します。

テキストフィールドに記入されている SQL 文をイベント番号1を指定して実行する

接続項目	接続関係
接続元コンポーネント	
(イベント発生コンポーネント)	
発生イベント	アクションイベント
接続先コンポーネント	■ローカル MySQL 管理 (ID:2)
起動メソッド	イベント番号を指定して SQL 文を実行する(String, int)
<引数0>	取得方法:メソッド戻り値
	コンポーネント:テキストフィールド(ID:3, KEY:SQL 文)
	メソッド/値:テキストを取得する
<引数1>	取得方法:固定值
	メソッド/値:1

実行結果をテーブルに設定する

接続項目	接続関係
接続元コンポーネント	
(イベント発生コンポーネント)	
発生イベント	データ生成イベント
接続先コンポーネント	■テーブル (ID:5)
起動メソッド	テーブルデータを設定する(PFObjectTable)
<引数>	説明:テーブルデータ
	取得方法:イベント内包
	メソッド/値:イベント対象データ
イベント番号	1

SQL 文実行結果は、「ローカル MySQL 管理」複合コンポーネントが発生するデータ生成イベントの内包 データとして取得します。SQL 文実行時にイベント番号として1を指定しましたので、データ生成イベ ントのイベント番号も1となることにご注意ください。

## 確認 接続関係と配置は以下のようになります。



図 25 SQL 文実行機能の追加(コンポーネント接続関係)

ワンポイント ーコメントー

指定したイベント番号が分かるように、イベント番号をコメントとして記述しておくと後で見るとき に便利です。起動メソッドにコメントを記述するには、コメントを記述する接続先コンポーネントを右 クリック- [起動メソッドコメント編集...] とクリックします。

# 操作 画面を作成しましょう。

- ① 画面編集ボタンをクリックします。
- ② 配置方法を [手動配置] に切り替えます。
- ③ 画面編集画面で「テキストフィールド」コンポーネントをフレームに追加します。
- ④ 画面編集画面で「ボタン」コンポーネントをフレームに追加します。
- ⑤ 画面編集画面で「テーブル」コンポーネントをフレームに追加します。

■ フレーム [ID:1] (KEY:"フレーム1") ■ ヨーカルMySQL管理 [ID:2] (KEY:"ローカルMy		5	フレーム [ID:1] (KEY:	"フレーム1")	
<ul> <li>□ テキストフィールド (ID.3) (KEY: SOL文")</li> <li>□ ボタン (ID.4) (KEY: "SOL文要(")</li> <li>□ ボタン (ID.5) (KEY: "テーブル5")</li> </ul>	select * from mac	hine	SQL文集	87	
	<del>テ</del> ーブル情報 テーブル名称 テーブル構成	フィールド名 (英勢	(作成 決字) デ	更新         崩除           一夕型         主キー	-12002
	ーローカルMySQLI ドライバ	1 f7រដ្ឋh0 1 f7ត/JBR ខ្លែរីឆ com.mysql.cj.jdbc.Driver	ソート データベース作成 ダンフ生成	クエリー覧 ・コピー・操作コマンド設定 :MySQLWySQL Server 8.0V	型友換一覧 i bin\mysqldump.exe

図 26 SQL 文実行機能の追加(コンポーネントの画面配置)

⑥ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

### 操作 アプリケーションの動作を確認しましょう。

- ① 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② [接続]ボタンをクリックし、「test」データベースに接続します。
- ③ [一覧設定]ボタンをクリックし、テーブル一覧の中に「machine」テーブルが含まれていることを 確認します。
- ④ テキストフィールドに「select \* from machine」と SQL 文を入力します。
- ⑤ [SQL 文実行]ボタンをクリックし、行データが空のテーブルが表示されることを確認します。

840					8			^
select * from mac	hine		SQL文集					
entryDay I	name id							
テーブル情報								
テーブル名称			作成	更新	削除		一覧設定	
=								
	フィールド	名(英数字)	7	-9型	<u>主</u> +-		machine	
τ = )/ν(φρχ	フィールド	*名(英数字)	Ţ	<u>- 夕型</u>	主キ~		machine	
τ = <i>Ο</i> Λνίφοχ	フィールド 1fī追加 1f	"名(英數字) 行削除		<u>- 夕型</u>	主キー クエリー覧		machine 型変换→】	Ë
ナーブル(#### ローカルMySQL	フィールド 1f5追加 1f 段定	"名(英数字) 行削除	デ ソート ータベース作成	<u>- 夕型</u> ・コピー・	<u>主キー</u> クエリー覧 陳作コマンド	· 	machine 型変換一】	£
デージル(#### ローカル(MySQL) ドライバ	フィールド 11行追加 11 設定 com.mysql.cj.jdbc	"名(英数字) 行削除 .Driver	デ ソート タベース作成 タンブ生成	ータ型 ・コピー・ MySQLW	主キー クエリー覧 操作コマンド IySQL Server	[] [] [設定	machine 型変換→1 i\mysqldur	۲
ナーブル(#px _ ローカルMySQL ドライバ サーバ	コィールド 1fī追加 1f 飲定 zom.mysql.cj.jdbc jdbc:mysql://locali	「石(英数字) 「育解除 たDriver host/	デ ソート タベース作成 ダンブ生成 MySQL起動	- 夕型 ・コピー・ MySQLW 1 FilesWy	主キー クエリー覧 操作コマンド IVSQL Server SQL/MySQL 3		machine 型変換→ J \\mysqldur 3.0\bin\mys	۲ ۳p.exe sql.exe
テークル(#0x _ _ _ ーローカルMySQL 」 ドライバ サーバ ユーザ名	フィールド 1行追加 1f 設定 com.mysql.cj.jdbc jdbc:mysql://localf root	名(英数字) 行酬除 Driver host/	デ       ソート       ータベース作成       ダンブ生成       MySQL超動       ータベース作成	- 夕型 ・コピー・i iMySQLW 1 FilesWyi	タエリー覧 操作コマンド SQL Server SQLWJSQL :	·····································	machine 型変換一ず \\mysqldur 3.0\bin\mys	۳p.exa sql.exa
- ローカルMySQL를 ドライバ サーバ ユーザ名 バスワード	フィールド 1fī追加 1f 設定 com.mysql.cj.jdbc jdbc:mysql://localt root	名(英数字) (Tall)& .Driver host/ デー	デ - タベース作成 ダンブ生成 MySQL超数 - タベース作成	- 夕型 · 그 년 - · · ·MySQLW h FilesWy: create dat	主キー クエリー覧 操作コマンド IVSQL Server SQL WySQL st tabase if not	t :設定 r 8.0\bin Server 8 exists	machine 型変換~1 i\mysqldur 3.0\bin\mys	۲ mp.exi sql.exi
- ローカルMySQL ドライバ サーバ ユーザ名 バスワード データベース名	フィールド 1fīi追加 1f 没定 com.mysql.cj.jdbc jdbc:mysql://locall root ●●●● test	-名(英数字) (7副9) :Driver host/ デデ	デ       ソート       ータベース作成       ダンフ生成       MySQL起動       ータベース作成       ータベース指定	- 夕型 ・コピー・・ ・MySQLW 1 FilesWy: create dat use	主キー クエリー覧 操作コマンド SQL Server SQLWySQL : tabase if not	·····································	machine 型変換→】 \\mysqldur 3.0\bin\mys	۳p.exi sql.exi

図 27 SQL 文の実行

⑥ 確認後、アプリケーションを終了します。

#### 2.6. 複数行一括追加機能の追加

データベーステーブルに複数のレコードを一括して追加する機能を追加します。4

#### 追加するコンポーネント

- テキストフィールド
- ボタン

#### 追加する機能

- 1. ボタンをクリックしたら、
- 2. テキストフィールドで指定したデータベーステーブルに対して
- 3. テーブルに記入されている複数の行データを一括追加する。

### 操作 コンポーネントを追加し、接続しましょう。

- ① 必要なコンポーネントを追加します。
   作業領域で右クリック-[コンポーネント追加] [画面構成部品] [テキスト] [テキストフィールド] とクリックします。
   作業領域で右クリック-[コンポーネント追加] [画面構成部品] [ボタン] [ボタン]
   とクリックします。
- ② 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。
   [ボタン] コンポーネントの[text]を「複数行追加」と設定します。
   [テキストフィールド] コンポーネントの[ComponentKey]を「追加先テーブル名」と設定します。

<sup>4</sup> 購買管理 04.mzax

③ 下の表に示すようにコンポーネントを接続します。

指定したテーブルに複数の行データを一括追加する

接続項目	接続関係
接続元コンポーネント	
(イベント発生コンポーネント)	■ ハタン(ID:7, NET:複数行追加)
発生イベント	アクションイベント
接続先コンポーネント	■ローカル MySQL 管理 (ID:2)
起動メソッド	指定したテーブルに複数行を一括追加する
	(String, PFObjectTable)
<引数0>	取得方法:メソッド戻り値
	コンポーネント:テキストフィールド(ID:6,
	KEY:追加先テーブル名)
	メソッド/値:テキストを取得する
<引数1>	取得方法:メソッド戻り値
	コンポーネント:テーブル(ID:5)
	メソッド/値:テーブルデータを取得する

確認 接続関係と配置は以下のようになります。



図 28 複数行一括追加機能の追加(コンポーネント接続関係)

# 操作 画面を作成しましょう。

- ① 画面編集ボタンをクリックします。
- ② 画面編集画面で「テキストフィールド」コンポーネントをフレームに追加します。
- ③ 画面編集画面で「ボタン」コンポーネントをフレームに追加します。

フレーム (ID:1) (KEY:"フレーム1")	716 ID-10////////////////////////////////////	
<ul> <li>□ ローカルMySOL管理 [ID-2] (KEY: TローカルMy</li> <li>□ テネストフィールド [ID-3] (KEY: SOL文)</li> <li>□ ボタン [D-4] (KEY: SOL文用)</li> <li>□ テ・ブル [ID-5] (KEY: デーブル5)</li> <li>□ テキストフィールド [ID-6] (KEY: 5動加先テー:</li> <li>□ ボタン [ID-7] (KEY: 7戦対行通知)</li> </ul>	select * from machine SOLX#/7	2882(F)3830
	テーブル指輯       テーブル場成     フィールド名(英数字)       デーブル場成	Mik         一覧設定           主キー
	1(〒追加 1(〒副除 ソート	クエリー覧

図 29 複数行一括追加機能の追加(コンポーネントの画面配置)

④ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

### 操作 アプリケーションの動作を確認しましょう。

- ① 実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② [接続]ボタンをクリックし、「test」データベースに接続します。
- ③ テーブルにレコードを追加します。
   テーブル上で右クリック-[追加]-[行]-[複数行…]とクリックし、3行と設定します。
- ④ テーブルにデータを入力します。
   「entryDay」と「name」に任意にデータを入力します。
   「id」は自動設定されますので、入力の必要はありません。
- ⑤ テーブル名として「machine」を設定し、[複数行追加]ボタンをクリックします。

複数行う追加			
談定			
談定			
設定			
設定			
設定			
设定			
line			
〔换一覧			
qldump.ex			
Files\MySQL\MySQL Server 8.0\bin\mysql.ex			
s			

図 30 複数行一括追加

⑥ [SQL 文実行]ボタンをクリックし、結果を確認します。
 id が自動で設定されていることが確認できます。

<b>2</b> .							1000		×
elect * from ma	chine			SQL文実	۲ <b>ד</b>	machi <mark>n</mark> e		複数行	追加
entryDay 2023/04/03 9:00: 2023/04/03 9:00: 2023/04/03 9:00:	00 ABC 00 DEF 00 GHI	e id	1 2 3						
テーブル情報									
テーブル名称	machine			作成	更新	削除		一覧設定	
テーブル構成	71-1	レド名(英数	字)	デー	タ型		- [	machine	
	0 entryDay			java.util.Date					_
-	1 name			java.lang.String					
	1 (Tiệ tr	1 (78)82		1		011			7
	1 (7)200	1 行削除		]		クエリ・	→覧	型変換一覧	<b>č</b>
ローカルMySQL	1行追加 設定	1 行道明殊	- ソート データベー	-ス作成	· コピー・	クエリ・ ・ 操作コマン	→覧 ノド設定	型変換一覧	ī
ローカルMySQL ドライバ	1 行追加 設定 	1 行前财余 jdbc.Driver	ソート データベー ダンプ	- ス作成 生成	コピー WySQL	クエリ・ ・操作コマン MySQL Ser	→覧 ノド設定 ver 8.0\b	型変換→腎 Din\mysqldun	ī np.ex
ローカルMySQL ドライバ サーバ	1行追加 設定 com.mysql.cj. jdbc:mysql://l	1 行首即余 jdbc.Driver pcalhost/	ソート データベー ダンプ MySQ	- ス作成 生成 起動	・コピー ·MySQL 1 FilesW	クエリ・ ・操作コマン MySQL Ser IySQLMySQ	→覧 ノド設定 ver 8.0\b DL Serve	型変換一覧 pin\mysqldun r 8.0\bin\mys	ž np.ex :ql.ex
ローカルMySQL ドライバ サーバ ユーザ名	1 行追助0 設定 	1行削除 jdbc.Driver ocalhost/	ソート データベー タンフ MySQ データベー		コピー :MySQL 1 FilesW create d	クエリ・ ・操作コマン MySQL Ser lySQLWySQ atabase if r	一覧 ノド設定 ver 8.0\b DL Serve not exists	型変換一覧 pin\mysqldun rr 8.0\bin\mys	ī np.ex
ローカルMySQL ドライバ サーバ ユーザ名 パスワード	1 行流動加 設定 	1 行動除 jdbc.Driver ocalhost/	ソート       データベー       ダンプ       MysQ       データベー	-ス作成 生成 -ス作成 -ス作成	・コピー :MySQL 1 FilesW create d	クエリ・ ・操作コマン MySQL Ser NySQLWySC atabase if r	→覧 ノド設定 ver 8.0\b DL Serve not exists	型変換一手 pin\mysqldun r 8.0\bin\mys	ž np.ex :ql.ex
ローカルMySQL ドライバ サーバ ユーザ名 バスワード データベース名	1 行流血加 設定 jdbc:mysql.cj. jdbc:mysql://l root ●●●● test	1行削除 jdbc.Driver ocalhost/	データベー ダンフ MySQ データベー データベー	ス作成 生成 起動 ス作成 ス指定	・コピー・ ·MySQL' 1 Files\M create d use	クェリ・ ・操作コマン MySQL Ser IySQLWySC atabase if r	→覧 > ド設定 ver 8.0\b DL Serve not exists	型変換一手 pin\mysqldun r 8.0\bin\mys	ž np.exe

図 31 複数行追加結果の確認

⑦ アプリケーションを終了します。

#### 2.7. バックアップ機能の追加

データベースのバックアップには、MySQLのダンプ機能を利用します。<sup>5</sup>

追加するコンポーネント

- メニュー
- メニューアイテム
- ファイル選択
- テキストファイル出力

#### 追加する機能

- 1. バックアップメニューを選択したら、
- 2. 出力先ファイルを選択し、
- 3. ダンプ文字列を書き出す。
- 4. ダンプ文字列を書き出した後、出力先ファイルを閉じる。

### 操作 コンポーネントを追加し、接続しましょう。

必要なコンポーネントを追加します。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [メニュー] [メニュー] とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [メニュー] [メニューアイテム] とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [ダイアログ] [ファイル選択] とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [入出力] - [ファイル] [テキストファイル出力] とクリックします。

 ② 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。
 [メニュー] コンポーネントの[text]を「データベース」、
 [メニューアイテム] コンポーネントの[text]を「バックアップ…」、
 [ファイル選択] コンポーネントの[ComponentKey]を「ダンプ出力ファイル選択」、
 [テキストファイル出力] コンポーネントの[ComponentKey]を「ダンプファイル出力」と
 設定します。

<sup>&</sup>lt;sup>5</sup> 購買管理 05.mzax
③ 下の表に示すようにコンポーネントを接続します。

バックアップメニューを選択したらファイル選択ダイアログを表示する

接続項目	接続関係
接続元コンポーネント	- $        -$
(イベント発生コンポーネント)	$\sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{i$
発生イベント	アクションイベント
接続先コンポーネント	■ファイル選択(ID:10, KEY:ダンプ出力ファイル選択)
起動メソッド	単数 Save 用ファイル選択ダイアログを表示する
	(Component)
<引数>	説明:親コンポーネント
	取得方法:固定值
	メソッド/値:なし

選択したファイルを設定する●

接続項目	接続関係
接続元コンポーネント	■ファノル選択(ID:10 KEV:ダンプルキファノル選択)
(イベント発生コンポーネント)	
発生イベント	データ選択イベント
接続先コンポーネント❶	■テキストファイル出力(ID:11, KEY:ダンプファイル出力)
起動メソッド	setFile(File)
<引数>	取得方法:イベント内包
	メソッド/値:イベント対象データ
イベント番号	1

ダンプファイルを生成する2

接続先コンポーネント❷	■ローカル MySQL 管理(ID:2)
起動メソッド	ダンプ生成()
イベント番号	1

ファイルを指定して、ダンプ文字列を書き出す●

接続項目	接続関係
接続元コンポーネント	
(イベント発生コンポーネント)	
発生イベント	データ設定イベント
接続先コンポーネント❶	■テキストファイル出力(ID:11, KEY:ダンプファイル出力)
起動メソッド	ファイルオブジェクトを指定して文字列を書き出す
	(File, String)
<引数 0>	説明:ファイルオブジェクト
	取得方法:メソッド戻り値
	コンポーネント:テキストファイル出力(ID:11,
	KEY:ダンプファイル出力)
	メソッド/値:ファイルオブジェクトを取得する
<引数1>	説明:書き出す文字列
	取得方法:イベント内包
	メソッド/値:イベント対象データ

ファイルを閉じる❷

接続先コンポーネント❷	■テキストファイル出力(ID:11, KEY:ダンプファイル出力)
起動メソッド	ファイルを閉じる()



接続関係と配置は以下のようになります。



## 図 32 バックアップ機能の追加(コンポーネント接続関係)

# 操作 面面を作成しましょう。

- ① 画面編集ボタンをクリックします。
- ② 「メニュー」コンポーネントをフレームに追加します。左側のコンポーネント一覧から「フレーム」
   コンポーネントを右クリックー[コンポーネント追加] [メニュー] とクリックします。
- ③ 「メニューアイテム」コンポーネントをメニューに追加します。左側のコンポーネント一覧から
   「メニュー」コンポーネントを右クリックー[コンポーネント追加] [メニューアイテム] と
   クリックします。
- ④ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

フレーム [ID:1] (KEY:"フレーム1")		フレーム [ID:1] (KEY:"フレーム1")	
<ul> <li>→ J ×ニューアイテム [D:9] (KEY:7) 590アン</li> <li>□ → カルMySQL管理 [D:2] (KEY:7) → カバMy</li> <li>□ テキストフィール下 [D:3] (KEY:SQL文)</li> <li>□ ボタン [D:4] (KEY:7SL文実行)</li> <li>□ テキストフィールド [D:6] (KEY:505)</li> <li>□ テキストフィールド [D:6] (KEY:505)</li> <li>□ テキストフィールド [D:6] (KEY:505)</li> <li>□ ボタン [D:7] (KEY:4507)</li> </ul>	select * from machine entryDay name id	SQL艾夷行	神旗数行注直加
	テーブル情報 テーブル名称 テーブル痛成 フィールド名(英	作成         更新         副数           数字)         データ型         主キー	一覧設定
	11771£80 117781984	- ソート - クエリー覧	

図 33 バックアップ機能の追加(メニューの追加)

# 操作 アプリケーションの動作を確認しましょう。

- ① 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② [接続]ボタンをクリックし、「test」データベースに接続します。
- ③ メニューから [データベース] [バックアップ…] と選択し、バックアップファイルを作成します。 保存先フォルダを「AP\_DATA¥Production」にし、ファイル名を「backup. sql」として保存します。

バックアップ	SOL文実行 複数行追加	
itryDay nam	e id	
	🛃 保存	
	保存: 늘 Production	<ul> <li>Ø 📴 🖽 •</li> </ul>
- ブル名称 - ブル構成	Image: Sector 2     Image: Sector 2       Image: Sector 2     Image: Sector 2	7667-
1		

図 34 バックアップファイルの作成(ダンプ出力)

④ アプリケーションを終了します。

# 操作 出力先ファイル(backup.sql)の内容を確認しましょう。

- ① エクスプローラを起動し、「backup. sql」ファイルを保存したフォルダを表示します。
- ② 「backup. sql」ファイルをダブルクリックし、ファイルを開きます。

データベースを作成するための一群の SQL 文が記述されていることが確認できます。

backup.sql × +	-		×
ファイル 編集 表示			<b>(</b> )
<pre> MySQL dump 10.13 Distrib 8.0.32, for Win64 (x86_64) Host: localhost Database: test</pre>			
<pre>/*i50503 SET character_set_client = umcharacter_set_client '/; /*i50503 SET character_set_client = utf8mb4 */; CREATE TABLE `machine` (     `entryDay` datetime DEFAULT NULL,     `name` text,     `id` int NOT NULL AUTO_INCREMENT,     PRIMARY KEY (`id`) ) ENGINE=InnoDB AUTO_INCREMENT=4 DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_0900_ai_ci;</pre>			
行1、列1 100% Windows (CRLF)	UTF-	8	



③ 確認後、ファイルを閉じます。

### 2.8. データベース内容設定機能の追加

作成したバックアップファイルから、データベースの内容を設定する機能を追加します。6

### 追加するコンポーネント

- メニューアイテム
- ファイル選択
- テキストファイル入力

## 追加する機能

- 1. 内容設定メニューを選択したら、
- 2. バックアップファイルを選択し、
- 3. データベースの内容を設定する。
- 4. ダンプ実行後、バックアップファイルを閉じる。

## 操作 コンポーネントを追加し、接続しましょう。

① 必要なコンポーネントを追加します。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [メニュー] [メニューアイテム] とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [ダイアログ] [ファイル選択] とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [入出力] - [ファイル] [テキストファイル入力]とクリックします。

② 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。
 [メニューアイテム] コンポーネントの[text]を「内容設定…」、
 [ファイル選択] コンポーネントの[ComponentKey]を「バックアップファイル選択」、
 [テキストファイル入力] コンポーネントの[ComponentKey]を「バックアップファイル入力」
 と設定します。

<sup>&</sup>lt;sup>6</sup> 購買管理 06.mzax

③ 下の表に示すようにコンポーネントを接続します。

内容設定メニューを選択したらファイル選択ダイアログを表示する

接続項目	接続関係
接続元コンポーネント	- メニューマイテル (ID:12 KEV:内容設定 )
(イベント発生コンポーネント)	
発生イベント	アクションイベント
接続先コンポーネント	■ファイル選択(ID:13, KEY:バックアップファイル選択)
起動メソッド	単数 Open 用ファイル選択ダイアログを表示する
	(Component)
<引数>	説明:親コンポーネント
	取得方法:固定值
	メソッド/値:なし

ファイルオブジェクトを指定して読み込む●

接続項目	接続関係	
接続元コンポーネント	■ フ _ ノ u '翠扫 (ID:10 VEV: v ケ フ プ フ _ ノ u '翠扫)	
(イベント発生コンポーネント)	■ファイル選択(ID:IS, KEI:ハックテッフファイル選択)	
発生イベント	データ選択イベント	
接続先コンポーネント❶	■テキストファイル入力 (ID:14,	
	KEY:バックアップファイル入力)	
起動メソッド	ファイルオブジェクトを指定して読み込む(File)	
<引数>	説明:読み込むファイルオブジェクト	
	取得方法:イベント内包	
	メソッド/値:イベント対象データ	
イベント番号	1	

ファイルを閉じる❷

接続先コンポーネント❷	■テキストファイル入力	(ID:14,
		KEY:バックアップファイル入力)
起動メソッド	ファイルを閉じる()	
イベント番号	1	

データベースの内容を設定する

接続項目	接続関係
接続元コンポーネント	■テキストファイル入力 (ID:14,
(イベント発生コンポーネント)	KEY:バックアップファイル入力)
発生イベント	データ生成イベント
接続先コンポーネント	■ローカル MySQL 管理 (ID:2)
起動メソッド	ダンプ実行(Object,int)
<引数0>	取得方法:イベント内包
	メソッド/値:イベント対象データ
<引数1>	取得方法:固定值
	メソッド/値:0



接続関係と配置は以下のようになります。



図 36 データベース内容設定機能の追加(コンポーネント接続関係)

# 操作 画面を作成しましょう。

- ① 画面編集ボタンをクリックします。
- ② 「メニューアイテム」コンポーネントをメニューに追加します。左側のコンポーネント一覧から 「メニュー」コンポーネントを右クリックー[コンポーネント追加] - [メニューアイテム] と クリックします。
- ③ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

* ■ フレーム [ID:1] (KEY:"フレーム1") □		5	7 レーム [ID:1] (KEY:"]	フレーム1")	
メニューアイテム [ID:0] (KEY: 1) ックアッ	select * from machine		SQL文集	ífī	複数行追加
<ul> <li>→ メニュ - /イテム [ID:12] (KEY: Y6/8355)</li> <li>→ ホルメSQL Weige ID:03 (KEY: Y80L)</li> <li>テキストフィールド [ID:3] (KEY: SQL)</li> <li>ボタン [ID:4] (KEY: *SQL)</li> <li>テーブル [ID:5] (KEY: *テーブル5")</li> <li>テキストフィールド [ID:6] (KEY: *5)</li> <li>デキストフィールド [ID:6] (KEY: *30)</li> </ul>	entryDay	name id			
	- テーブル情報 - テーブル情報 - テーブル名称		作成	更新	一覧設定
	テーブル構成	フィールド名(英	敬字〉   デ	- 夕型   主キー	
	テーブル構成	フィールド名(英 加 1行詞除	<u></u> デ	- 夕型 主キー クエリー覧	型实换一覧
	テーブル構成	フィールド名(英 加 1行嗣除	数字) デ ソート データペース作成	- 夕型 主キー クエリー覧 ・コピー・操作コマンド設定	
	テーブル構成 11行追 -ローカル/MySOL設定 ドライバ 50m.m	フィールド名(英 加 1行鋼線 1ysql.cj.jdbc.Driver	数字) デ	ーク型 主キー クエリー覧 ・コビー・操作コマンド設定 MySQL Server 8.01	型皮换一覧
	テーブル構成 1行達 ローカルMSGLI設定 ドライバ 20mm サーバ jdbcm ユーザ名 root	フィールド名(英 加 1行網線 nysql.cj.jdbc.Driver nysql/ilocalhost/	数字) デ データペース作成 ダンブ生成 MySQL起動	- 夕型 主キー クエリー覧 ・コピー・操作コマンド設定 MySQLMySQL Server 8.010 1 FilesMySQLWySQL Server	整定换一覧 型定换一覧 pin\mysqldump.exe er 8.0\bin\mysql.exe
	テーブル構成 11行組 ローカルMySQLL設定 ドライバ 20mm サーバ jdbcm ユーザ名 root パスワード	フィールド名(英 加 1行酬除 nysql.cj.jdbc.Driver nysql.//localhost/	<ul> <li>数字) デ</li> <li>ソート</li> <li>データペース作成</li> <li>ダンブ生成</li> <li>MySQL超動</li> <li>データペース作成</li> <li>データペース作成</li> </ul>	- ク型 主キー クエリー覧 ・コピー・操作コマンド設定 MySQLMySQL Server 8.00 1 FilesMySQL Server 8.00 1 FilesMySQL Server	型实换一页 型实换一页 pin\mysqldump.exe er 8.0\bin\mysql.exe s
	テーブル構成 1151年 ローカルMySOLL設定 ドライバ 20mm サーバ jdbcm ユーザ名 root バスワード データベース名 test 接抜 デー	フィールド名(英 加 1行嗣除 nysql.cjjdbc.Driver nysql//localhost/ タベース新規作成	<ul> <li>         (ソート)         データペース作成         ダンフ生成         データペース作成         データペース作成         データペース作成         データペース指定         MySQL&amp;ア         MySQL&amp;ア         MySQL&amp;ア         MySQL&amp;ア         MySQL&amp;ア         MySQL&amp;P         MySQL&amp;P         MySQL&amp;P         MySQL         MySQL</li></ul>	- ク型 主キー クエリー覧 ・コビー・操作コマンド設定 MySQLMySQL Server 8.0t 1 FilesMySQLMySQL Server create database if not exist use quit	발호뉴-턴 Din\mysqldump.exe s s

図 37 データベース内容設定機能の追加(メニューへの追加)

それでは、新しく「test1」という名前のデータベースを作成し、その内容を、前節で作成したバック アップファイルで設定してみます。

操作 「test1」データベースを「backup. sq1」を使って作成しましょう。

① 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。

② [データベース名] に「test1」と入力し、[データベース新規作成]ボタンをクリックします。

ータベース									^
elect * from mac	hine			SQL文実	ít 🗌			複数	行追加
entryDay	na	ame	id						
テーブル情報 テーブル名称 🗌				作成	更新	首眼余		一覧設定	.))
1 00 X XX 800 A			1.5						
テーブル構成	7	ィールド名(	英数字)	<u> </u>	- タ型	<u></u>	-		
テーブル構成	フ 1 行注意知口	ィールド名( 117首明線	英数字) ソート	<u> </u>	- 夕型 ク型	<u>主</u> キー	- 覧	型変換一	·覧
テーブル構成 ローカルMySQL	フ 1行追加 役定	ィールド名( 1行調除	英数字) ソート データイ	デ·	- 夕型 ・ コピー・1	主キ- クェリー 衆作コマン	- 覧 ド設定	型変換一	·覧
テーブル構成 ローカルMySQL ドライバ	フ 11行達加 役定 	ィールド名( 1行調除 ql.cj.jdbc.Drive	英数字) ソート データイ ダン	デ・ ベース作成 マフ生成	- タ型 ・ コピー・i MySQLW	」 主キー クエリー 衆作コマン ySQL Serv	 覧 ド設定 er 8.0\b	型変換-	•覧 Jmp.ex
テーブル構成 ローカルMySQL ドライバ サーバ	フ 1 行注動加 設定 20m.mysc jdbc:mysc	イールド名( 1行謝除 ql.cj.jdbc.Drive ql://ocalhost/	英数字) ソート データイ yr MyS	デ・ マース作成 マ生成 au起動	- 夕型 ・コピー・ i MySQLW i FilesWyS	シェリー クエリー 衆作コマン ySQL Serv SQLMySQI	- 覧 ド設定 er 8.0\b Server	型変換- In\mysqldu 8.0\bin\m	•覧 imp.ex ysql.ex
テーブル構成 ローカルMySQL ドライバ サーバ ユーザ名	フ 1行追加 設定 jdbc:mysc root	イールド名( 1行調除 ql.cj.jdbc.Drive ql.//localhost/	英数字) ソート データイ MyS データイ	デ・ マース作成 マ生成 :30.22動 マース作成	- タ型 ・コピー・ i MySQLW 1 FilesWyS create dat	主キー クェリー 衆作コマン ySQL Serv GQL/MySQI abase if no	ー 覧… ド設定 - er 8.0\b _ Server ot exists	型変換- in\mysqldu - 8.0\bin\m	•覧 Imp.ex ysql.ex
テーブル構成 ローカルMySQL ドライバ サーバ ユーザ名 バスワード データベーフ	フ 1行注血加 没定 20m.mysc jdbc:mysc root	ィールド名( 1行調除 ql.cj.jdbc.Drive ql://localhost/	英数字) ソート データイ ダン MyS データイ データイ	デ・ マース作成 マリ生成 このL起動 マース作成 マース指定	- 夕型 ・コピー・1 <sup>i</sup> MySQLW 1 FilesWyS create dat use	クエリー 衆作コマン ySQL Serv GQLWySQI abase if no	- 覧 ド設定 er 8.0\b _ Server ot exists	型変摘- in\mysqldu 8.0\bin\m	•覧 Jmp.ex ysql.ex

図 38 データベース test1 の新規作成

 ③ メニューから、[データベース] - [内容設定…] と選択し、前節で作成したバックアップファイル 「backup. sql」を指定します。

2	– o ×	
データベース バックアップ… 内容設定…	SQL文実行 視题(行道加) name id	
	▲ 開<	×
	ファイルの場所(I): 📁 Production	v 👌 📴 🛄 •
テーブル信報 テーブル名称 テーブル構成		JUE2-
1f ローカルMySQL設定 ドライパ ion サーバ idb/ ユーザ名 roo	PC 27/1/2011: [Dackup.sq]	
パスワード		- 取消
データベース名 test 横続 う	1 「クベース解焼作成 MySQL終了 quit	

図 39 データベース内容設定

④ アプリケーションを終了します。

## 操作 「test1」データベースが作成されていることを確認しましょう。

- ① 実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② 現在表示されているテーブルを削除します。
   テーブル内で右クリック-[テーブル] [列] [全削除] とクリックします。
- ③ [接続]ボタンをクリックし、「test1」データベースに接続します。
- ④ [一覧設定]ボタンをクリックし、machine テーブルが作成されていることを確認します。
- ⑤ [SQL 文実行]ボタンをクリックして、テーブルの内容が設定されていることを確認します。

<b>ニ</b> ョ データベース							-		×
elect * from m	achine			3QL文実	行			*复数行	行追加
entryDay	name	id							
023/04/03 9:00	0:00 ABC		1						
023/04/03 9:00	0:00 DEF		2						
023/04/03 9:00	D:00 GHI		3						
テーブル情報 テーブル名称	machine			作成	更新	削除	<	→覧設定	>
テーブル構成	フィール	ド名(英数:	字)	デー	タ型	主+-		machine	
	0 entryDay			java.ut	il.Date				
	1 name		13	java.lan	g.String				
	2 id	2 id		java.lang.Integer 🛛 🛃					
	1173830	1 行削除	V-F	]		クエリー	<b>覧</b>	型変換→	覧
ローカルMySQ	1行追加 L設定	1 ftäure	ト データベー	·ス作成	・コピー・扌	クエリー 衆作コマン	覧 ド設定	 型変換→	Ë
ローカルMySQ ドライノ	11行追加 11行追加 L設定 <sup>デ</sup> _com.mysql.cj.jc	1 行前明全 bc.Driver	ソート データベー ダンプ	- ス作成 生成	・コピー・扌 :WySQLW	クエリー 兼作コマン ySQL Serv	覧 ド設定 er 8.0\t	型変換→ 」 vin\mysqldu	覧 mp.ex
ローカルMySQ ドライノ サーノ	1行追加 L設定 <sup>で</sup> 「「jdbc:mysql.//loc	1 行前時注 Ibc.Driver ;alhost/	ソート データベー ダンプ: MySQL	- ス作成 生成 起動	・コピー・扌 iMySQLW	クエリー 衆作コマン ySQL Serv sQLMySQ	覧 ド設定 er 8.0\b _ Serve	型変換→ pin\mysqldu r 8.0\bin\my	覧 mp.ex sql.ex
ローカルMySQ ドライノ サーノ ユーザ名	1行追加 LL設定 <sup>5</sup> com.mysql.cj.jc jdbc:mysql://loo G root	1 行前除 lbc.Driver :alhost/	ソート データペー タンプ! MySQL データペー	- ス作成 生成 起動 - ス作成	・コピー・ł :MySQLW 1 FilesWyS create dat	クエリー 衆作コマン ySQL Serv SQL/MySQ abase if n	覧 ド設定 er 8.0\t Serve ot exists	型変換→ pin\mysqldu r 8.0\bin\my	覧 mp.ex sql.ex
ローカルMySQ ドライ) サーJ ユーザネ パスワー	1 行动動加 L設定 <sup>5</sup> _ com.mysql.cj.jc jdbc:mysql://lor 3 _ root F* ●●●●	1 行前除 lbc.Driver :alhost/	ソート データベー タンプ: MySQL データベー	-ス作成 生成 起動 -ス作成 -ス指定	・コピー・扌 :WySQLW 1 FilesWyS create dat:	クエリー 衆作コマン ySQL Serv sQL\MySQ abase if n	覧 ド設定 er 8.0\t L Serve ot exists	型変換→ Din\mysqldu r 8.0\bin\my	覧 mp.ex sql.ex
ローカルMySQ ドライノ サーノ ユーザネ パスワー   データベース4	11行追加 LL設定 <sup>5</sup> com.mysql.cj.jc <sup>7</sup> jdbc.mysql.//lod <sup>5</sup> root <sup>6</sup> <sup>6</sup> <sup>6</sup> <sup>6</sup> <sup>6</sup> <sup>6</sup> <sup>6</sup> <sup>6</sup> <sup>6</sup> <sup>6</sup>	1 行前除 lbc.Driver alhost/	ソート データベー タンプ: MySQL データベー データベー	-ス作成 生成 起動 -ス作成 -ス指定 01 終マ	・コピー・ iMySQLM 1 FilesMyS create dat: use	クエリー 衆作コマン ySQL Serv GQLWySQ abase if n	覧 ド設定 er 8.0\b L Serve ot exists	型変換→ Din\mysqldu r 8.0\bin\my	覧 mp.ex sql.ex

図 40 データベース内容設定結果の確認

⑥ アプリケーションを終了します。

#### 2.9. データベーステーブル編集機能の追加

データベーステーブルは、UPDATE 文を実行することで編集することができます。

UPDATE <テーブル名> SET <カラム名> = <値>, … WHERE 制約条件;

しかしながら、複数のレコードやフィールドを編集する場合には、エクセルのような表を書き換える 方が容易です。「データベースアクセス」コンポーネントには、「検索結果を用いたテーブル更新処理」 という以下の機能があります。

- 1. SELECT 文で取得したテーブルを内部に保持し、
- 2. そのテーブルを更新すると、自動的に MySQL データベーステーブルを更新する。

ここでは、この「検索結果を用いたテーブル更新処理」機能を利用し、アプリケーションにデータベ ーステーブル編集機能を追加することにします。下の図は、この機能の概略を示したものです。



図 41 データベーステーブル編集機能の概略

SELECT 文で取得した検索結果を、「データベースアクセス」コンポーネント内部に保持するとともに、 アプリケーションの GUI テーブルに設定します。ユーザの GUI テーブル編集結果に応じて、「データベ ースアクセス」コンポーネントが保持しているテーブルに値を設定します。すると、「データベースア クセス」コンポーネントによる MySQL データベーステーブルの更新が自動で行われます。 ここでは、データベースアクセスコンポーネントが持つ以下の3つのメソッドを使います。

メソッド	機能
startUpdate()	「検索結果を用いたテーブル更新処理」を開始します。
updateCell(Object,int,int)	「検索結果を用いたテーブル更新処理」の実行中に、「データベー
	スアクセス」コンポーネントが保持しているテーブルのセル値を
	設定します。
<pre>finishUpdate()</pre>	「検索結果を用いたテーブル更新処理」を終了します。

そのために、まず、「ローカル MySQL 管理」複合コンポーネントへ移動し、これらを公開メソッドとして指定します。これらのメソッドは日本語化されていませんので、メソッド選択ダイアログで「全メソッド対象」をチェックします。

## 操作 メソッドを公開しましょう。

- ① 「ローカル MySQL 管理」複合コンポーネントへ移動します。
- 「ローカル MySQL 管理」複合コンポーネントを右クリック- [公開メソッド設定…]をクリック します。
- ③ 「データベースアクセス」コンポーネントを右クリック- [公開メソッド設定…] をクリック します。
- ④ [全メソッド対象] チェックボックスをチェックし、一覧から「finishUpdate()」「startUpdate()」「updateCell(Object, int, int)」を選択し追加(公開化)ボタンをクリックします。続けて設定 ボタンをクリックします。



図 42 公開メソッドの設定

⑤ [公開メソッド設定]ダイアログを閉じ、アプリケーションビルダーのトップ画面に移動します。

それでは、データベーステーブル編集機能を追加します。7

# 追加するコンポーネント

- ダイアログ
- メニューアイテム
- テキストフィールド
- ボタン
- テーブル

## 追加する機能

- 1. テーブル編集メニューを選択したら、
- 2. データベースアクセスコンポーネントの「検索結果を用いたテーブル更新処理」を開始し、
- 3. ダイアログが表示される。
- 4. ダイアログ上のボタンをクリックするとテキストフィールドの SELECT 文を実行し、
- 5. 実行結果がダイアログ上の GUI テーブルに設定される。
- 6. GUI テーブルを編集すると、
- 7. データベースアクセスコンポーネントが内部に保持しているテーブルが更新される。
- 8. ダイアログを閉じると、
- 9. データベースアクセスコンポーネントの「検索結果を用いたテーブル更新処理」を終了し、
- 10. 編集していた GUI テーブルの内容をすべて削除する。

<sup>7</sup> 購買管理 07.mzax

## 操作 | コンポーネントを追加し、接続しましょう。

① 必要なコンポーネントを追加します。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [ダイアログ] - [ダイアログ] とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [メニュー] - [メニューアイテム] とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [テキスト] - [テキストフィールド]とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [ボタン] - [ボタン]とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [デーブル] - [テーブル]とクリックします。

2 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。
 [ダイアログ] コンポーネントの[title]を「テーブル編集」、
 [メニューアイテム] コンポーネントの[text]を「テーブル編集…」、
 [テキストフィールド] コンポーネントの[Columns]を「16」、[ComponentKey]を「SELECT 文」、
 [ボタン] コンポーネントの[text]を「テーブル取得」、
 [テーブル] コンポーネントの[ComponentKey]を「編集対象テーブル」
 と設定します。

③ 下の表に示すようにコンポーネントを接続します。

テーブル編集メニュ	ーを選択したら検索組	吉果を用いたテーブ	ル更新処理を開始する●
-----------	------------	-----------	-------------

接続項目	接続関係
接続元コンポーネント	- ノー フノニノ (ID:16 KEV: ニ_ ゴル 毎年 )
(イベント発生コンポーネント)	<mark>- アーユーアイ テム(ID-10, INLI-アーフル補未)</mark>
発生イベント	アクションイベント
接続先コンポーネント❶	■ローカル MySQL 管理 (ID:2)
起動メソッド	startUpdate()

ダイアログを表示する2

接続先コンポーネント❷	■ダイアログ(ID:15, KEY:テーブル編集)
起動メソッド	ダイアログを表示する()

ボタンをクリックするとテキストフィールドに入力された SELECT 文を実行する

接続項目	接続関係
接続元コンポーネント	■ ボタン (ID:18 KEV:テーブル取得)
(イベント発生コンポーネント)	<b>■</b> パダン(ID:16, KEI:) — クル取得)
発生イベント	アクションイベント
接続先コンポーネント	■ローカル MySQL 管理(ID:2)
起動メソッド	イベント番号を指定して SQL 文を実行する (String, int)
<引数0>	取得方法:メソッド戻り値
	コンポーネント : テキストフィールド(ID:17,
	KEY:SELECT 文)
	メソッド/値:テキストを取得する
<引数1>	取得方法:固定值
	メソッド/値:2

実行結果をテーブルに設定する

接続項目	接続関係
接続元コンポーネント	
(イベント発生コンポーネント)	
発生イベント	データ生成イベント
接続先コンポーネント	■テーブル (ID:19, KEY:編集対象テーブル)
起動メソッド	テーブルデータを設定する(PFObjectTable)
<引数>	説明:テーブルデータ
	取得方法:イベント内包
	メソッド/値:イベント対象データ
イベント番号	2

SELECT 文(SQL 文)を実行するときのイベント番号を2としましたので、「ローカル MySQL 管理」 複合コンポーネントから発生するデータ生成イベントの番号も2を指定しています。

テーブルを編集すると「データベースアクセス」コンポーネント内部のテーブルが更新される

接続項目	接続関係
接続元コンポーネント	■ニゴ』 (ID:10 KEV: 絙佐社会ニゴ』)
(イベント発生コンポーネント)	「「ノル (ID:19, NLI:袖未対象) ー ノル)
発生イベント	データ更新イベント
接続先コンポーネント	■ローカル MySQL 管理 (ID:2)
起動メソッド	updateCell(Object, int, int)
<引数0>	取得方法:イベント内包
	メソッド/値:更新データ
<引数1>	取得方法:メソッド戻り値
	コンポーネント:テーブル (ID:19, KEY:編集対象テーブル)
	メソッド/値:選択行の位置を取得する
<引数2>	取得方法:メソッド戻り値
	コンポーネント:テーブル (ID:19, KEY:編集対象テーブル)
	メソッド/値:選択列の位置を取得する
イベント番号	0

ダイアログを閉じると検索結果を用いたテーブル更新処理を終了する●

接続項目	接続関係
接続元コンポーネント	■ダイマログ(ID:15 KEV:ニ_ブル纪年)
(イベント発生コンポーネント)	
発生イベント	アクションイベント
接続先コンポーネント❶	■ローカル MySQL 管理(ID:2)
起動メソッド	finishUpdate()

GUI テーブルの内容を全て削除する2

接続先コンポーネント❷	■テーブル(ID:19, KEY:編集対象テーブル)
起動メソッド	全行列を削除する()

## 確認

接続関係と配置は以下のようになります。



図 43 データベーステーブル編集機能の追加(コンポーネント接続関係)

### 操作 | 画面を作成しましょう。

- 画面編集ボタンをクリックします。
- ② 「メニューアイテム」コンポーネントをメニューに追加します。左側のコンポーネント一覧から
   「メニュー」コンポーネントを右クリックー[コンポーネント追加] [メニューアイテム] と
   クリックします。
- ③ 左側のコンポーネント一覧から「ダイアログ」コンポーネントを右クリックー[コンポーネント
   追加] [テキストフィールド] とクリックします。
- ④ 左側のコンポーネント一覧から「ダイアログ」コンポーネントを右クリックー[コンポーネント
   追加] [ボタン] とクリックします。
- ⑤ 左側のコンポーネント一覧から「ダイアログ」コンポーネントを右クリックー[コンポーネント
   追加] [テーブル] とクリックします。
- ⑥ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。



図 44 データベーステーブル編集機能の追加(ダイアログの画面配置とメニューへの追加)

## 操作 アプリケーションの動作を確認しましょう。

- ① 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② [データベース名] に「test」と入力し、[接続]ボタンをクリックします。
- ③ メニューから [データベース] [テーブル編集...]を選択し、[テーブル編集] ダイアログを 表示させます。

データベース								
バックアッコ	1		SC	L文実行				複数行追加
内容設定… テーブル編集	name	ə id						
テーブル水和								
テーブル名称			0	d E	<b>F</b> 1 1 3	ALC: N		一般設定
		1. 1. 2. 744	58-12")	ポーク別		++-	T II	
テーフル構め	1 745	-ルト名 (典						
テーフル構め、	71-	-7/1-76 (94						
テーフル構め、 _	24-	-ルト名 (奥						
7 - J//#bx _	24-	-10 F 26 (A	・ テーブル編ま	L.			×	1
テーフル構成 _ _	24-		テーブル編集	E.		テーブル	× 取得	
テーフル構め、 -	1155800	-//12 (A	・ テーブル編集	K	-10	テーブル	× /取得	12天桥一段
テーフル袖bx - ローカルMySQL	1/ījātā 1/ījātā	1/78 (A	・ テーブル編集	5.	_)(	テーブル	× 取得	型实施一段
テーフル構成 一 ローカルMySOL3 ドライバ	1行jata 1行jata 設定 	1fjate Jdbc.Driv	テーブル編集	K		テーブル	× 取得	©実施→覧 ysqldump.e
テーフル構成 ローカルMySQL ドライバ サーバ	1 fījātā 1 fījātā 陳定 jdbc:mysql.cj jdbc:mysql./j	1 f7inte Jdbc.Driv localhost	テーブル編集	K		テーブル	× -取得	如果 如果 如果 如 如 如 一 知 二 二 一 二 二 一 二 二 二 二 二 二 二 二 二 二 二 二 二
テーフル構成 ローカルMySQL ドライバ サーバ ユーザ名	1 fījāta 1 fījāta 20m.mysql.cj jdbc.mysql.// root	1 find	・ テーブル編集	K		テーブル	× 和政律	di <del>x in fi</del> r iysqldump. Ubin'mysql.
テーフル構成 ローカルMySQL ドライバ サーバ ユーザ名 バスワード	1fšiato krz zom.mysql.cj jdbc:mysql.idi root	1 find	テーブル編集	7 16/27		テーブル	× 取得	¥末時一覧 ysqldump. bin\mysql.e
テージル構成 ローカルMySQL ドライバ サーバ ユーザ名 バスワード デー <u>タペース</u> 名	1fijato btž zom.mysql.cj jdbc.mysql.ll root test	コロア名(A	テープル編集	L 入指定 use		テーブル	× 取得	な水池一記 ysqidump.i bbintmysql.i

図 45 テーブル編集ダイアログの表示

④ [テーブル編集] ダイアログのテキストフィールドに「select \* from machine」と入力し、
 [テーブル取得]ボタンをクリックします。

### !!!注意事項!!!

取得したテーブルには、主キーのフィールドが含まれていなくてはなりません。

entryDay	name	id	
2023/04/03	ABC	1	
2023/04/03	. DEF	2	
2023/04/03.	. GHI	3	

図 46 テーブルの取得

⑤ テーブルを適当に編集し、ダイアログを閉じます。

select * fron	n machine		リレ相対待
entryDay	name	id	
2023/04/03	ABC-1	1	
2023/04/03	DEF-2	2	
2023/04/03.	GHI-3	3	

図 47 テーブルの編集

⑥ [SQL 文実行]ボタンをクリックして検索を実行し、データベーステーブルの内容が更新されている ことを確認します。

ニータベース				-	o x
elect * from ma	chine	SQL文実	87		複数行适力
entryDay 023/04/03 9:00 023/04/03 9:00 023/04/03 9:00	name id 00 ABC-1 00 DEF-2 00 GHI-3	1 2 3			
テーブル情報					
テーブル名称		flat	更新	RURE	一覧設定
2. 2.10 mpA	フィールト名(央京	(牛) ア	- 2 至	±+-	
	21-1/176 (48	<u>x+)                                     </u>	-72	<u></u> _	
	1(示直加) 1(示解除		->=	<u></u>	型实质一段
р- лимуsor	2イールド名(奥5 1行減加 1行線線 換定		- y型   ・コピー・揀介	<u></u> クエリー覧 Fコマンド設定	
ローカルMySQL ドライバ	1行通知 1行時間 1行通知 1行時間 検定 :com.mysql.cj.jdbc.Driver	ス作成 タンフ生成	・コピー・操作 ·MySQLWyS(	クエリー覧 Fコマンド設定 DL Server 8.00	シェア第一覧 ショート binimysqidump.exe
ローカルMySQL ドライバ サーバ	1行通知 1行解除 設定 jdbc.Driver jdbc.mysql./j.jdbc.Driver	ソート       データペース作成       ダンフ生成       MySOL設わ	・コピー・操作 ·MySQLWySQ 」FilesWySQL	クエリー覧 ミコマンド設定 DL Server 8.00 .WySQL Serv	ይገድው የአማት መስከት የአማት የአማት የአማት የአማት የአማት የአማት የአማት የአማ
ローカルMySQL ドライバ サーバ ユーザ名	1173880 1177688 IATE com.mysql.cj.jdbc.Driver jdbc.mysql./jlocalhost/ root	ソート データペース作成 ダンブ生成 MySOL級物 データペース作成	・コピー・挿作 MySQLWyS( 1 FilesWySQI create databa	クエリー靴 Fコマンド設定 DL Server 8.01 UMySQL Serv	발표해~원 binimysqidump.exe er 8.0/binimysqi.exe ts
ローカルMySQL ドライバ サーバ ユーザ名 バスワード	11751100 1177688 11751100 1177688 ISTE com.mysql.cj.jdbc.Driver jdbc:mysql.//localhost/ root ••••	ソート     データベース作成     ダンブ生成     MySOL超物     データベース作成     データベース作成     データベース作成     データベース	・コピー・挿作 MySQLW/SQL reate databa	クエリー靴 Fコマンド設定 DL Server 8.01 LWySQL Serv ase if not exist	발표폐-원 2 bin\mysqidump.exe er 8.0\bin\mysqi.exe ts
ローカルMySOL ドライバ サーバ ユーザ名 バスワード データペース名	1175abo 117688 Briz com.mysql.cj.jdbc.Driver jdbc:mysql.//localhost/ root eest	ソート     データベース作成     ダンフ生成     MySOL超数     データベース作成     データベース作成     データベース指定     WoON #27	・コピー・神行 ·MySQLWySQ h FilesWySQL create databa	クエリー靴 Fコマンド設定 DL Server 8.01 LWySQL Server	「加速期一覧」 「加速期一覧」 に た の し 加 小mysql dump.exe er 8.0 し 加 小mysql exe ち に り に り の し の し の し の の の の し の の の の の の の の

図 48 データベーステーブル更新結果の確認

⑦ 確認後、アプリケーションを終了します。

# 3. 発注依頼部の作成

データベース連携部に続いて、発注依頼部を作成します。

# 3.1. 購買管理データベースの作成

はじめに、購買管理データベースを作成します。

# 操作 kobai データベースを作成しましょう。

- ① 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② [データベース名] に「kobai」と入力し、[データベース新規作成]ボタンをクリックします。
- ③ メニューから [データベース] [内容設定…] を選択します。AP\_DATA¥Tutorial¥購買管理フォル ダ内の「kobai.sql」を指定します。
- ④ [接続]ボタンをクリックし「kobai」データベースに接続します。
- ⑤ [一覧設定]ボタンをクリックし、以下のデータベーステーブルが作成されていることを確認します。
  - irai(発注依頼)

項目	主キー	注文番号	注文日	発注部門コード	購買品コード	メーカ名
フィールド名	id	kobaiCD	iraibi	bumonCD	hinmokuCD	maker
データ型	Integer	String	Date	String	String	String

規格仕様	希望納期	発注数量	単価	金額
kikaku	kiboNouki	kiboSuryo	kiboTanka	kiboKingaku
String	Date	Integer	Double	Double

(仮)注文先コード	製品設備コード	検査の有無	部門承認
kiboKobaisakiCD	seihinCD	kiboKensa	shonin
String	String	Boolean	Boolean

#### kobai(購買管理)

項目	主キー	注文番号	注文先コード	検査の有無	納期回答	発注数量
フィールド名	id	kobaiCD	kobaisakiCD	kensa	nouki	suryo
データ型	Integer	String	String	Boolean	Date	Integer

発注単価	発注合計	納入日	受入担当者	完了日
tanka	kingaku	nonyubi	ukeireTanto	nouki
Double	Double	Date	String	Date

• keihi(経費管理)

	項目	主キー	注文番号	経費部門コード	費目コード
--	----	-----	------	---------	-------

フィールド名	id	kobaiCD	keihiBumonCD	himokuCD
データ型	Integer	String	String	String

• kobai\_mstr(購買マスター)

項目	主キー	購買品コード	中区分コード	品名	メーカ名	規格仕様
フィールド名	id	hinmokuCD	chuKubunCD	hinmei	maker	kikaku
データ型	Integer	String	String	String	String	String

単価
tanka
Double

• kubun\_mstr(区分マスター)

項目	主キー	大区分コード	大区分	中区分コード	中区分
フィールド名	id	daiKubunCD	daiKubun	chuKubunCD	chuKubun
データ型	Integer	String	String	String	String



• kobaisaki\_mstr(購買先マスター)

項目	主キー	注文先コード	注文先	指定納品書
フィールド名	id	kobaisakiCD	kobaisaki	yoshiki
データ型	Integer	String	String	String

▶ himoku\_mstr(費目マスター)

項目	主キー	費目コード	費目
フィールド名	id	himokuCD	himoku
データ型	Integer	String	String

• 5	seihin_	_mstr	(製品設備	マス	ター)

項目	主キー	製品設備コード	製品設備名
フィールド名	id	seihinCD	seihin
データ型	Integer	String	String

• bumon\_mstr(部門マスター)

項目	主キー	部門コード	部門名	承認者
フィールド名	id	bumonCD	bumon	shoninID
データ型	Integer	String	String	String

これらのテーブルのフィールドのうち、参照関係にあるものは同じ色で示してあります。

⑥ 確認後、アプリケーションを終了します。

## 3.2. 「発注依頼」複合コンポーネントの作成

発注依頼部は、複合コンポーネントとして作成することにします。

│ 操 作 │ 「発注依頼」複合コンポーネントを作成しましょう。

- ② 作成した複合コンポーネントをダブルクリックし、複合コンポーネントの中へ移動します。
- ③ コンポーネント名称を設定します。 ビルダー画面上部のコンポーネント名称欄に「発注依頼」と入力し【Enter】キーを押します。
- ④ コンポーネントキーを設定します。属性情報設定画面を表示し、[ComponentKey]を「発注依頼」と設定します。

WZ Platform アプリケーションビルター			
ファイル 編集 アブリケーション オブション	ン ヘルブ	コメント行検索	
アブリケーション名称			
アプリケーション	MZ Platform アプリケーションビルダー		
KEY:=	ファイル 編集 アプリナ・ション オブション ヘルブ	- IVA	ī検索 💌
=	コンボーネン 名称 発注依頼		🔁 🖻 🖄 📥
	1 発注依頼		
	ID:20 KEY:"発注体賴"		
(I <no name=""></no>			
ID:20 KEY:			
17V-4			
В.1 КЕҮ:"21-41"			
□□-カルMySQL管理 ID:2			
KEY:"ローカルMySQL管理"			
<u>■</u> ダイアログ			
ID:15 KEY:"テーブル編集"			
[アイコン凡例] ■アプリケー3			
実行 実行			
	【アイコン凡例】 <b>●</b> アプリケーション ● 「画面構成部品 ●   ウィンドウ	ジメニュー ジパネル ジ処理部品 創業合(画面)	構成) 🗾 複合(処理) 🛑 リモート
	実行 実行(設定可) 画面編集 帳票編集	ロード 挿入 保存 上書き保存 :	197 終了

図 49 複合コンポーネントの生成

#### 3.3. 初期コンポーネントの追加

この複合コンポーネントは、トップ階層にある「ローカル MySQL 管理」 複合コンポーネントとの間で、 イベントの送受信によるデータの交換を行います。このように、上位階層とイベントの送受信を行うこ とが想定される複合コンポーネントの場合、最初に以下のコンポーネントを追加しておくと便利です。

追加するコンポーネント

- フレーム
- イベント生成(受信用)
- イベント生成(送信用)
- サブルーチン(初期化処理用)
- サブルーチン(終了処理用)

## 操作 コンポーネントを追加し、コンポーネントキーを設定しましょう。

① 必要なコンポーネントを追加します。
 作業領域で右クリック-[コンポーネント追加] - [画面構成部品] - [ウィンドウ] [フレーム] とクリックします。
 作業領域で右クリック-[コンポーネント追加] - [処理部品] - [イベント] [イベント生成] とクリックします。これを受信用とします。
 作業領域で右クリック-[コンポーネント追加] - [処理部品] - [イベント] [イベント生成] とクリックします。これを送信用とします。
 作業領域で右クリック-[コンポーネント追加] - [処理部品] - [サブルーチン] [サブルーチン]とクリックします。これを初期化処理用とします。
 作業領域で右クリック-[コンポーネント追加] - [処理部品] - [サブルーチン] [サブルーチン]とクリックします。これを終了処理用とします。

 ② 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。

 [フレーム] コンポーネントの[title]を「発注依頼」、[ComponentKey]を「発注依頼」、
 受信用として追加した [イベント生成] コンポーネントの[ComponentKey]を「受信イベント」、
 送信用として追加した [イベント生成] コンポーネントの[ComponentKey]を「送信イベント」、
 初期化処理用として追加した [サブルーチン] コンポーネントの[ComponentKey]を「初期化処理」、
 終了処理用として追加した [サブルーチン] コンポーネントの[ComponentKey]を「終了処理」
 と設定します。

■ ※注佐頼	
1D:20	
KEY:"発注依頼"	
170-4	
ID : 20-1	
LKEY:"発注依頼"	
「一イベント生成	
ID : 20-2	
LKEY:"受信イベント	
「一イベント生成	
ID : 20-3	
LKEY:"送信イベント	
サブルーチン	
ID : 20-4	
LKEY:"初期化処理"	
ーチン	
ID : 20-5	
/    ) / , リシカ    7 わり 7用	

## 図 50 初期コンポーネントの追加

## 3.4. 公開メソッドおよびコンポーネント接続の初期設定

トップ階層から起動されることが想定されるメソッドを、最初に公開メソッドとして設定しておきます。

コンポーネント	公開するメソッド	目的
フレーム	フレームを表示する()	
	フレームを閉じる()	
イベント生成	データ生成イベントの発生	トップ階層の「ローカル MySQL 管理」複合コ
(受信イベント)	(Object, int)	ンポーネントから発生するデータ生成イベ
		ントを伝達するため
サブルーチン	処理を呼び出す()	トップ階層から初期化処理を起動するため
(初期化処理)		
サブルーチン(終了処理)	処理を呼び出す()	トップ階層から終了処理を起動するため

## 操作 メソッドを公開しましょう。

- ① 「発注依頼」複合コンポーネントを右クリック- [公開メソッド設定…]をクリックします。
- 「フレーム」コンポーネントを右クリック-[公開メソッド設定…]をクリックします。
- ③ 一覧から「フレームを表示する()」「フレームを閉じる()」を選択し追加(公開化)ボタンを クリックします。続けて設定ボタンをクリックします。
- ④ 「イベント生成(受信イベント)」コンポーネントを右クリック--[公開メソッド設定…]を クリックします。
- ⑤ 一覧から「データ生成イベントの発生(Object, int)」を選択し追加(公開化)ボタンを クリックします。続けて設定ボタンをクリックします。
- ⑥ 「サブルーチン(初期化処理)」コンポーネントを右クリック- [公開メソッド設定…]を クリックします。
- ⑦ 一覧から「処理を呼び出す()」を選択し追加(公開化)ボタンを
   クリックします。続けて設定ボタンをクリックします。
- ⑧ 終了処理と区別するため、公開メソッド名を変更します。
   「cal1()-->処理を呼び出す()」を右クリック-[公開メソッド名変更…]をクリックし、
   「初期化処理を呼び出す」と入力します。

- ⑨ 「サブルーチン(終了処理)」コンポーネントを右クリック- [公開メソッド設定…]を クリックします。
- 一覧から「処理を呼び出す()」を選択し追加(公開化)ボタンをクリックします。
   続けて設定ボタンをクリックします。
- ① 初期化処理と区別するため、公開メソッド名を変更します。
   「cal1()-->処理を呼び出す()」を右クリック-[公開メソッド名変更…]をクリックし、
   「終了処理を呼び出す」と入力します。
- 12 [公開メソッド設定] ダイアログを閉じます。



図 51 公開メソッドの設定コンポーネント接続の初期設定

また、以下のコンポーネントから発生するイベントの接続先を設定しておきます。<sup>8</sup>

- イベント生成(送信イベント)
   トップ階層の「ローカル MySQL 管理」複合コンポーネントが持つ「イベント番号を指定して SQL 文 を実行する(String, int)」メソッドへの引数を伝達するため、データ生成イベントを上位階層へ伝 播させるようにしておきます。ここで使用するイベントは、必ずしもデータ生成イベントでなくて も構いません。
- サブルーチン(終了処理)
   終了処理としてフレームを閉じるようにしておきます。

# 操作 コンポーネントを接続しましょう。

下の表に示すようにコンポーネントを接続します。

接続項目	接続関係
接続元コンポーネント	■イベント生式 (ID:20_2 KEV:送信イベント)
(イベント発生コンポーネント)	■1ハント王成(ID-20-3, NET-医信1ハンド)
発生イベント	データ生成イベント
接続先コンポーネント	■発注依頼(ID:20, KEY:発注依頼)
起動メソッド	イベントを伝播させる(PFEvent)
<引数>	説明:対象イベント
	取得方法:イベント

データ生成イベントを上位階層へ伝播させる

フレームを閉じる

接続項目	接続関係
接続元コンポーネント	■サゴリチン。(ID:20_5 KEV: 終了加理)
(イベント発生コンポーネント)	■ リノルーテン(ID:20-5, NET:終了処理)
発生イベント	アクションイベント
接続先コンポーネント	■フレーム(ID:20-1, KEY:発注依頼)
起動メソッド	フレームを閉じる()

<sup>&</sup>lt;sup>8</sup> 購買管理 08.mzax

確認 接続関係と配置は以下のようになります。

発注依頼 ID:20 KEY:"発注依頼"			
<mark></mark>			
<mark>  イベント生成</mark>  D:20-2  KEY:"受信イベント"			
● イベント生成 ID:20-3 KEY:"送信イベント"	データ生成イベント	イベントを伝播させる	■ 発注依頼 ID:20 KEY:"発注依頼"
U サブルーチン ID:20-4 KEY:"初期化処理"			
<mark>] サブルーチン</mark> ID : 20-5 KEY : "終了処理"	アクションイベント	フレームを開じる	□フレーム ID:20-1 KEY:"発注依頼"

図 52 コンポーネント接続の初期設定

# 3.5. トップ階層におけるイベント接続先の初期設定

アプリケーションビルダーのトップ階層へ移動し必要となるイベントの接続先を設定しておきます。?

操作 コンポーネントを接続しましょう。

- ① アプリケーションビルダーのトップ画面に移動します。
- 下の表に示すようにコンポーネントを接続します。

初期化処理を呼び出す

接続項目	接続関係
接続元コンポーネント	
(イベント発生コンポーネント)	
発生イベント	アプリケーション開始イベント
接続先コンポーネント	■発注依頼(ID:20, KEY:発注依頼)
起動メソッド	初期化処理を呼び出す()

終了処理を呼び出す

接続項目	接続関係
接続元コンポーネント	■ フプリケ
(イベント発生コンポーネント)	
発生イベント	アプリケーション終了イベント
接続先コンポーネント	■発注依頼(ID:20, KEY:発注依頼)
起動メソッド	終了処理を呼び出す()

<sup>&</sup>lt;sup>9</sup> 購買管理 09.mzax

イベント番号を指定して SQL 文を実行する

接続項目	接続関係
接続元コンポーネント	■ 祭汁佐語 (ID:20 KEV: 祭汁佐語)
(イベント発生コンポーネント)	■ 宪注104 程(ID·20, KET·宪注124程)
発生イベント	データ生成イベント
接続先コンポーネント	■ローカル MySQL 管理(ID:2)
起動メソッド	イベント番号を指定して SQL 文を実行する(String, int)
<引数0>	取得方法:イベント内包
	メソッド/値:イベント対象データ
<引数1>	取得方法:イベント内包
	メソッド/値:イベント番号

データ生成イベントの発生を起動する

接続項目	接続関係
接続元コンポーネント	■ローナ』 MySOL 答理 (ID:2)
(イベント発生コンポーネント)	■ローガル MySQL 官理(ID.2)
発生イベント	データ生成イベント
接続先コンポーネント	■発注依頼(ID:20, KEY:発注依頼)
起動メソッド	データ生成イベントの発生(Object,int)
<引数0>	説明:生成データ
	取得方法:イベント内包
	メソッド/値:イベント対象データ
<引数1>	説明:イベント番号
	取得方法:イベント内包
	メソッド/値:イベント番号

## 確認 接続関係と配置は以下のようになります。



図 53 トップ階層におけるイベントの接続先と起動メソッド

### 3.6. 発注依頼フレームのメニュー登録

発注依頼フレームをメニューから起動できるようにします。10

#### 追加するコンポーネント

- メニュー
- メニューアイテム

### 追加する機能

- 1. 発注依頼メニューを選択したら、
- 2. 発注依頼フレームが表示される。

操作 コンポーネントを追加し、接続しましょう。

- 必要なコンポーネントを追加します。
   作業領域で右クリック-[コンポーネント追加] [画面構成部品] [メニュー] [メニュー]
   とクリックします。
   作業領域で右クリック-[コンポーネント追加] [画面構成部品] [メニュー] [メニューアイテム] とクリックします。
- ② 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。
   [メニュー] コンポーネントの[text]を「購買管理」、
   [メニューアイテム] コンポーネントの[text]を「発注依頼…」と設定します。
- 下の表に示すようにコンポーネントを接続します。

発注依頼メニューを選択したら発注依頼フレームを表示する

接続項目	接続関係
接続元コンポーネント	- Jー - マノニノ (ID:00 KEV: & け休福 )
(イベント発生コンポーネント)	■ メニューアイナム(ID-22, NET·宪注1枚粮) 
発生イベント	アクションイベント
接続先コンポーネント	■発注依頼(ID:20, KEY:発注依頼)
起動メソッド	フレームを表示する()

<sup>&</sup>lt;sup>10</sup> 購買管理 10.mzax
## 確認 接続関係と配置は以下のようになります。



図 54 発注依頼フレームのメニュー登録(コンポーネント接続)

## 操作 画面を作成しましょう。

- 画面編集ボタンをクリックします。
- ② 「メニュー」コンポーネントをフレームに追加します。左側のコンポーネント一覧から「フレーム」
   コンポーネントを右クリックー[コンポーネント追加] [メニュー] とクリックします。
- ③ 「メニューアイテム」コンポーネントをメニューに追加します。左側のコンポーネント一覧から
   「メニュー」コンポーネントを右クリックー[コンポーネント追加] [メニューアイテム] と
   クリックします。
- ④ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

「フレーム [ID:1] (KEY:"フレーム1") - ↓メニュー [ID:8] (KEY:"データベース")			7	レーム [ID:1] (KEY:"フレー	·La1")	
メニューアイテム [ID:9] (KEY:"バックア・ メニューアイテム [ID:12] (KEY:"内容設定	select * from mai	chine		SQL文実行		複数行追加
	entryDay	name	id Id			
<ul> <li>→ ニューアイテム ID 22 (KEY: %EX # # # # # # # # # # # # # # # # # # #</li></ul>	- テーブル債種 テーブルる称 - テーブル構成	771 1f5iábo	-ルド名(英計 11行前8条	14.41 更 (年4.41) データ3 (ソート)	新 副称 型 主キー クエリー覧 -	-2022
	ローカルMySQL	版定		データベース作成・コt	ビー・操作コマンド設定	
	ドライバ	com.mysql.c	.jdbc.Driver	ダンプ生成 /My:	SQL/MySQL Server 8.0	bin\mysqldump.exe
	サーハフーザ名	jdbc:mysql://	ocalhost/	MySQL起動 1 Fil	es/MySQL/MySQL Serv	er 8.0\bin\mysql.exe
	パスワード	1001		データベース作成 crea	ite database if not exist	s
	データベース名	kobai		データベース指定use		
	接続	データペーン	新規作成	MySQLife 7 quit		
	A second					

図 55 発注依頼フレームのメニュー登録(メニューの追加)

## 操作 アプリケーションの動作を確認しましょう。

- ① 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② メニューから [購買管理] [発注依頼…]を選択し、[発注依頼] フレームが表示されることを 確認します。
- ③ メインウィンドウを終了し [発注依頼] フレームが閉じることを確認します。

### 3.7. 参照機能の作成と登録機能作成の準備

発注依頼部では、kobai データベースへアクセスし、irai テーブルの検索、レコードの登録、削除、 更新、そして発注依頼の伝票発行を行います。その概観は以下の通りです。

<b>約</b> 発注依頼	۹.						
		参照	登録	削除 注:	文番号一覧設定	۲) (۲	票発行…
注文番号	注文日	発注部門コード	購買品コード	発注数量	単価	金額	部門承認
A1	2011/11/17	S1	P1	5	10,000	50,000	(77)

#### 図 56 発注依頼部概観

SQL 文を作成するときには、あらかじめ定型文を用意しておき、必要に応じて文字列の代入や挿入を 行うようにしておくと便利です。発注依頼部のテーブルでは、kobaiCD(注文番号)、iraibi(注文日)、 bumonCD(発注部門コード)、hinmokuCD(購買品コード)、kiboSuryo(発注数量)、kiboTanka(単価)、 kiboKingaku(金額)、shonin(部門承認)のフィールドを使います。したがって、注文番号を指定してレ コードを検索するには以下の定型文を用意し、必要に応じて「KOBAICD」の部分を入れ替えます。

select kobaiCD, iraibi, bumonCD, hinmokuCD, kiboSuryo, kiboTanka, kiboKingaku, shonin from irai where kobaiCD='KOBAICD'

まず、参照機能の作成と登録機能作成の準備を行います。11

## 追加するコンポーネント

- ラベル
- ボタン(参照用)
- ボタン(登録用)
- テーブル(画面表示用)
- テーブル(登録用)
- 文字列格納変数(SQL 定型文を設定し、「KOBAICD」を検索対象の値で置き換えるために使う)
- テーブル格納変数
- サブルーチン(検索結果をテーブルに設定する処理を記述する)

#### 追加する機能

- 1. [参照]ボタンをクリックすると、
- 2. 文字列格納変数コンポーネントに SQL 定型文を設定し、
- 3. 「KOBAICD」を画面表示用テーブルの0行0列のセル値で置き換え、
- 4. その SQL 文をイベント番号 3 で実行し、
- 5. 取得した検索結果を、画面表示用と登録用の2つのテーブルに設定する。

<sup>&</sup>lt;sup>11</sup> 購買管理 11.mzax

### 操作 コンポーネントを追加し、接続しましょう。

- ① 「発注依頼」複合コンポーネントへ移動します。
- 必要なコンポーネントを追加します。 作業領域で右クリック- [コンポーネント追加] - [画面構成部品] - [テキスト] - [ラベル] とクリックします。 作業領域で右クリック- [コンポーネント追加] - [画面構成部品] - [ボタン] - [ボタン] とクリックします。これを参照用とします。 作業領域で右クリック- [コンポーネント追加] - [画面構成部品] - [ボタン] - [ボタン] とクリックします。これを登録用とします。 作業領域で右クリック-「コンポーネント追加]-「画面構成部品]-「テーブル]-「テーブル] とクリックします。これを画面表示用とします。 作業領域で右クリック- [コンポーネント追加] - [画面構成部品] - [テーブル] - [テーブル] とクリックします。これを登録用とします。 作業領域で右クリック-「コンポーネント追加]-「処理部品]-「変数]-「文字列格納変数] とクリックします。 作業領域で右クリック- [コンポーネント追加] - [処理部品] - [変数] - [テーブル格納変数] とクリックします。 作業領域で右クリック- [コンポーネント追加] - [処理部品] - [サブルーチン] - [サブルーチン] とクリックします。

③ 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。

[ラベル] コンポーネントの[text]を「select kobaiCD, iraibi, bumonCD, hinmokuCD, kiboSuryo, kiboTanka, kiboKingaku, shonin from irai where kobaiCD = 'KOBAICD'」、
[ComponentKey]を「SQL 定型文」、
参照用として追加した[ボタン] コンポーネントの[text]を「参照」、
登録用として追加した[ボタン] コンポーネントの[text]を「登録」、
画面表示用として追加した[テーブル] コンポーネントの[ComponentKey]を「画面表示用」、
登録用として追加した [テーブル] コンポーネントの[ComponentKey]を「登録用」、
[サブルーチン] コンポーネントの[ComponentKey]を「参照データ取得」
と設定します。

下の表に示すようにコンポーネントを接続します。

参照ボタンをクリックすると文字列格納変数に SELECT SQL 定型文を設定する●

接続項目	接続関係
接続元コンポーネント	■ ギカヽ, (ID:20_9 KEV:会昭)
(イベント発生コンポーネント)	
発生イベント	アクションイベント
接続先コンポーネント❶	■文字列格納変数(ID:20-12)
起動メソッド	文字列を設定する(イベント発生なし)(String)
<引数>	取得方法:メソッド戻り値
	コンポーネント:ラベル(ID:20-7, KEY:SQL 定型文)
	メソッド/値:ラベルのテキスト文字列を取得する

## 画面表示用テーブルの0行0列の値を返す❷

接続先コンポーネント❷	■テーブル (ID:20-10, KEY:画面表示用)
起動メソッド	指定セルの値を取得する(int, int)
<引数0>	取得方法:固定值
	メソッド/値:0
<引数1>	取得方法:固定值
	メソッド/値:0

KOBAICD を画面表示用テーブルの0行0列の値に置き換える❸

接続先コンポーネント€	■文字列格納変数(ID:20-12)
起動メソッド	正規表現に適合する最初の文字列を置換する
	(String, String)
<引数0>	取得方法:固定值
	メソッド/値:KOBAICD
<引数1>	取得方法:メソッド処理結果
	メソッド/値:指定セルの値を取得する

SQL 文をイベント番号3で実行する4

接続先コンポーネント❹	■イベント生成 (ID:20-3, KEY:送信イベント)
起動メソッド	データ生成イベントの発生(Object, int)
<引数0>	説明:生成データ
	取得方法:メソッド戻り値
	コンポーネント:文字列格納変数(ID:20-12)
	メソッド/値:文字列を取得する
<引数1>	説明:イベント番号
	取得方法:固定值
	メソッド/値 3

イベントを参照データ取得サブルーチンに渡して処理を呼び出す

接続項目	接続関係
接続元コンポーネント	■イベント生式 (ID:20_2 KEV: 马信イベント)
(イベント発生コンポーネント)	
発生イベント	データ生成イベント
接続先コンポーネント	■サブルーチン(ID:20-14, KEY:参照データ取得)
起動メソッド	イベントを渡して処理を呼び出す(PFEvent)
<引数>	説明:イベント
	取得方法:イベント
イベント番号	3

取得した検索結果を画面表示用テーブルに設定する●

接続項目	接続関係
接続元コンポーネント	■サゴリニエン(ID:20_14 KEV:会昭ゴニク取得)
(イベント発生コンポーネント)	■ リブルー / ブ (10·20-14, NE1·参照 / 一 文 取得)
発生イベント	データ生成イベント
接続先コンポーネント❶	■テーブル (ID:20-10, KEY:画面表示用)
起動メソッド	テーブルデータを設定する(PFObjectTable)
<引数>	説明:テーブルデータ
	取得方法:イベント内包
	メソッド/値:イベント対象データ

取得した検索結果を参照用テーブルに設定する
❷

接続先コンポーネント❷	■テーブル(ID:20-11, KEY:登録用)
起動メソッド	テーブルデータを設定する(PFObjectTable)
<引数>	説明:テーブルデータ
	取得方法:イベント内包
	メソッド/値:イベント対象データ

確認

接続関係と配置は以下のようになります。



図 57 参照機能作成と登録機能作成準備(コンポーネント接続図)

## 操作 | 画面を作成しましょう。

- 画面編集 ボタンをクリックします。
- ② 配置方法を [手動配置] に切り替えます。
- ③ 画面編集画面で「ボタン」コンポーネントおよび「テーブル」コンポーネントをフレームに 追加します。



## 図 58 参照機能作成と登録機能作成準備(画面配置)

閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

## 操作 テーブルに検索結果を設定しましょう。

- ① 実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② [接続]ボタンをクリックし、「kobai」データベースに接続します。
- ③ メニューから、[購買管理] [発注依頼…] と選択し、[発注依頼]フレームを表示します。
- ④ 画面表示用テーブル上で右クリックし、[テーブル] [テーブル作成…] と選択し、
   1行1列のテーブルを作成します。列型は java. lang. String のままとします。

		4	·照 登錄
テーブル・	テーブル作成 行 列 サイズ フォント 編集可否 行高 列幅 前景色 背景色 選択時の色	→ → → →	作成 空のテーブルを作ります。 行数 1 列数 1 デフォルトの列型 java.lang.String 適用 取消
	選択方法 グリッド線 グリッド色 行ヘッダ	* *	

図 59 画面表示用テーブルの作成

⑤ [参照]ボタンをクリックします。検索結果が2つのテーブルに設定されます。

影 発注依頼	<u> 新</u> 発注依頼	
	kobaiCD iraibi bumonCD hinmokuC <	林登 深登 0 , ,

図 60 検索結果の設定

画面表示用のテーブルの列名を適切な日本語に設定し、列幅を調整します。また、iraibi(注文日)の表示形式を日付のみに設定します。さらに、画面表示用と登録用の2つのテーブルに1行を追加します。

操作 面面表示用テーブルを編集しましょう。

画面表示用の列名を下の表に示すように日本語で設定します。
 変更する列名上を右クリックー[列] - [列名…] をクリックします。

変更前	kobaiCD	iraibi	bumonCD	hinmokuCD	kiboSuryo	kiboTanka	kiboKingaku	shonin
変更後	注文番号	注文日	発注部門	購買品	発注数量	単価	金額	部門承認
			コード	コード				

② それぞれの列幅を調整します。

- ③ iraibi(注文日)の列型を日付に変更します。
   列名上を右クリック- [列] [列型] [日付] をクリックします。
- ④ iraibi(注文日)の表示形式を日付のみに設定します。
   列名上を右クリック- [列] [表示形式(日付)] [表示形式] [日付のみ] をクリックします。



図 61 テーブルの列名および表示属性設定と行追加

⑤ 画面表示用と登録用のテーブルに1行追加します。
 テーブル上を右クリックー[テーブル] - [行] - [追加] - [一行] とをクリックします。

⑥ 画面表示用と登録用のテーブルの shonin(部門承認)の列型を論理に変更します。
 列名上を右クリックー [列] - [列型] - [論理] をクリックします。

画面編集画面でレイアウトを整えます。また、登録用テーブルは画面から削除しておきます。

操作 | 画面表示用テーブルのレイアウトを整え、登録用テーブルを削除しましょう。

- ① アプリケーションを終了します。
- ② 画面編集ボタンをクリックします。
- ③ レイアウトを整えます。
- ④ 登録用テーブルを削除します。

登録用テーブル上を右クリックー[コンポーネント削除]とクリックします。



図 62 画面レイアウトの調整

⑤ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

参照データ取得後のサブルーチンの処理を修正します。12

## 追加するコンポーネント

• 比較演算(>)

### 修正する機能

- 1. 参照データ取得サブルーチンのうち、画面表示用と登録用の2つのテーブルに検索結果を設 定する処理を削除する。
- 2. データ生成イベントの対象データ(検索結果)をテーブル格納変数に設定し、
- 3. その行数が0より大きければ、第0行のデータを画面表示用テーブルの第0行に設定する。

# 操作 コンポーネントを追加し、接続しましょう。

- ① 「発注依頼」複合コンポーネントへ移動します。
- ② 「サブルーチン(参照データ取得)」コンポーネントから接続されているメソッドをいったん全て 削除します。アクションイベント上で右クリックー[イベント処理削除]とクリックします。
- ③ 必要なコンポーネントを追加します。
   作業領域で右クリック-[コンポーネント追加] [処理部品] [条件制御] [比較演算(>)]
   とクリックします。

<sup>&</sup>lt;sup>12</sup> 購買管理 12.mzax

下の表に示すようにコンポーネントを接続します。

データ生成イベントのイベント対象データ(検索結果)をテーブル格納変数に設定する●

接続項目	接続関係
接続元コンポーネント	■サゴルーチン(ID:20_14 KEV:会昭データ取得)
(イベント発生コンポーネント)	
発生イベント	データ生成イベント
接続先コンポーネント❶	■テーブル格納変数(ID:20-13)
起動メソッド	テーブルを設定する(PF0bjectTable)
<引数>	説明:設定するテーブル
	取得方法:イベント内包
	メソッド/値:イベント対象データ

テーブル格納変数に格納されているテーブルの行数が0より大きいかを演算する2

接続先コンポーネント❷	■比較演算(>) (ID:20-15)
起動メソッド	数値変換/左右オペランド設定後、
	演算を行う(String,String)
<引数0>	説明:左オペランド
	取得方法:メソッド戻り値
	コンポーネント:テーブル格納変数(ID:20-13)
	メソッド/値:行数を取得する
<引数1>	説明:右オペランド
	取得方法:固定值
	メソッド/値:0

行数が0より大きければ(演算結果が true ならば)第0行のデータを取得する●

接続項目	接続関係
接続元コンポーネント	■ 比
(イベント発生コンポーネント)	■ L1牧,與身(ン) (10·20-13)
発生イベント	処理完了イベント
接続先コンポーネント❶	■テーブル格納変数(ID:20-13)
起動メソッド	行データリストを位置指定で取得する(int)
<引数>	説明:指定位置
	取得方法:固定值
	メソッド/値:0
イベント番号	1

テーブル格納変数に格納されている第0行のデータを画面表示用テーブルの第0行に設定する2

接続先コンポーネント❷	■テーブル (ID:20-10, KEY:画面表示用)
起動メソッド	指定行に行データをリスト形式で設定する
	(PFObjectList, int)
<引数0>	取得方法:メソッド処理結果
	メソッド/値:行データリストを位置指定で取得する
<引数1>	取得方法:固定值
	メソッド/値:0
イベント番号	1

確認



図 63 参照データ取得後の処理の修正

## 3.8. 登録機能の作成

発注依頼の登録には、INSERT 文(SQL 文)を利用しても構いませんが、ここでは「ローカル MySQL 管理」 複合コンポーネントが持つ「指定したテーブルに複数行を一括追加する(String, PFObjectTable)」メソ ッドを使うことにします。<sup>13</sup>

## 追加する機能

- 1. [登録]ボタンをクリックすると、
- 2. 画面表示用テーブルの行データを登録用テーブルの第0行に設定し、
- 3. データ更新イベントを使ってテーブル名(irai)とテーブルデータを送信する。
- トップ階層で、「発注依頼」複合コンポーネントから発生するデータ更新イベントを「ローカル MySQL 管理」複合コンポーネントに接続し、「指定したテーブルに複数行を一括追加する (String, PF0bjectTable)」を起動する。

<sup>&</sup>lt;sup>13</sup> 購買管理 13.mzax

## 操作 コンポーネントを接続しましょう。

① 下の表に示すようにコンポーネントを接続します。

登録ボタンをクリックすると画面表示用テーブルの行のデータを取得する●

接続項目	接続関係
接続元コンポーネント	■ ギカン (ID:20_0 KEV:発得)
(イベント発生コンポーネント)	■ バダン(ID:20-9, NEI:豆球)
発生イベント	アクションイベント
接続先コンポーネント❶	■テーブル(ID:20-10, KEY:画面表示用)
起動メソッド	指定行の行データをリスト形式で取得する(int)
<引数>	取得方法:固定值
	メソッド/値:0

取得した画面表示用テーブルの行のデータを登録用テーブルの第0行に設定する❷

接続先コンポーネント❷	■テーブル(ID:20-11, KEY:登録用)
起動メソッド	指定行に行データをリスト形式で設定する
	(PFObjectList, int)
<引数0>	取得方法:メソッド処理結果
	メソッド/値:指定行の行データをリスト形式で取得する
<引数1>	取得方法:固定值
	メソッド/値:0

データ更新イベントでテーブル名(irai)とテーブルデータを送信する

接続先コンポーネント❸	■イベント生成(ID:20-3, KEY:送信イベント)
起動メソッド	データ更新イベントの発生(Object,Object,int)
<引数0>	説明:対象データ
	取得方法:固定值
	メソッド/値:irai
<引数1>	説明:更新データ
	取得方法:メソッド戻り値
	コンポーネント:テーブル(ID:20-11, KEY:登録用)
	メソッド/値:テーブルデータを取得する
<引数 2>	説明:イベント番号
	取得方法:固定值
	メソッド/値:0

データ生成コンポーネントのデータ更新イベントを上位階層へ伝播する

接続項目	接続関係
接続元コンポーネント	■ イベント仕式 (ID:20_2 KEV:送信 イベント)
(イベント発生コンポーネント)	■1ハンド主成(ID-20-3, NET-医信1ハンド)
発生イベント	データ更新イベント
接続先コンポーネント	■発注依頼 (ID:20, KEY:発注依頼)
起動メソッド	イベントを伝播させる(PFEvent)
<引数>	取得方法:イベント

確認 接続関係と配置は以下のようになります。



## 図 64 登録機能の作成(「発注依頼」複合コンポーネント内)

- ② アプリケーションビルダーのトップ画面に移動します。
- ③ 下の表に示すようにコンポーネントを接続します。

発注依頼複合コンポーネントのデータ更新イベントをローカル MySQL 管理複合コンポーネントに接続し 指定したテーブル(irai)に複数行を一括追加する

接続項目	接続関係
接続元コンポーネント	■祭汁佐頡 (ID:20 KEV:祭汁佐頡)
(イベント発生コンポーネント)	■光注版粮(ID:20, NLI:光注版粮)
発生イベント	データ更新イベント
接続先コンポーネント	■ローカル MySQL 管理 (ID:2)
起動メソッド	指定したテーブルに複数行を一括追加する
	(String, PFObjectTable)
<引数0>	取得方法:イベント内包
	メソッド/値:イベント対象データ
<引数1>	取得方法:イベント内包
	メソッド/値:更新データ

### 確認 接続関係と配置は以下のようになります。



図 65 登録機能の作成(トップ階層)

## 操作 アプリケーションの動作を確認しましょう。

- ① 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② [接続]ボタンをクリックし、「kobai」データベースに接続します。
- ③ メニューから、[購買管理] [発注依頼…] とクリックします。
- ④ 適当なデータを入力して[登録]ボタンをクリックします。

6 発注依頼	ą.						
					(	参照	登録
注文番号	注文日	発注部門コード	購買品コード	発注数量	単価	金額	部門承認
A1	2011/11/17	S1	P1	10	10,000	100.000	

#### 図 66 登録機能の動作確認1

 ⑤ メインウィンドウで、「select \* from irai」という SQL 文を実行し、登録がされていることを 確認します。

select * from irai	$\mathbf{>}$			SQL文実	Ť		複数行追
shonin kit	ooKensa se	ihinCD	kiboKobai	kiboKinga	aku kiboTanka	kiboSury	o kiboNouki
				100,0	000 10,000		10
テーブル構成	-	a 18.25 Ze	And a state of the			2000	
S. 1999,0000 -		<u></u>	央奴子)	<u> </u>	- 久型   王	<u>+-</u>	
		1行削除	<u>央</u> 敬子) ソート	, <u>7</u> ~	-夕型   主 クェ クェ	(手一)	型変換一覧
	1行追加 設定	1行削除	央数子) ソート	<u>,</u> マース作成・	- 夕型   主 クェ コビー・操作コ <sup>、</sup>	<u>キー</u> リー覧 マンド設定	型交換一覧
ローカルMySQLI ドライバ	1行追加 設定 20m.mysql.cj.	JUF名 G 1行酬除 jdbc.Drive	典数子) ソ−ト データ/ ダン	ス作成・ マース作成・ - <b>- 7</b> 生成	- 3型 主 クェ コビー・操作コ: 	<u>リー覧</u> マンド設定 Server 8.0\b	型変換一覧 in/mysqldump.
ローカルMySQL ドライバ サーバ	1行追加 設定 jdbc:mysql://i	JUF名 G 1行削除 jdbc.Drive ocalhost/	典数子) ソート データ∕ II ダン My:	· · · · · · · · · · · · · · · · · · ·	- タ型 主 クエ コピー・操作ココ WiySQLWiySQLSU	キー リー覧… マンド設定 Server 8.0\b SQL Server	型変換→覧 in\mysqldump. r 8.0\bin\mysql
ローカルMySQL ドライバ サーバ ユーザる	1行追加 款定 jdbc:mysql://i root	1 行聊除 jdbc.Drive ocalhost/	央叔子) ソート データ・ ダン アータ・ メー	 ベース作成 - ソプ生成 SQL短動 ペース作成	- タ型 主 クェ コピー・操作コ・ MySQLWySQL S 1 FilesWySQLWy create database	キー リー覧 マンド設定 Server 8.0\b SQL Server if not exists	型変換一覧 in\mysqldump r 8.0\bin\mysql
ローカルMySaL ドライバ サーバ ユーザ名 パスワード	1行追加 致定 com.mysql.cj. jdbc:mysql://l root	1 行御除 jdbc.Drive ocalhost/	央訳子) ソート データ・ ダン データ・ データ・ データ・ データ・	、 マース作成 パフ生成 SQL短動 マース作成 マース作成	- タ型 主 クェ コピー・操作コ・ MySQLWySQL S 1 FilesWySQLWy create database	キー リー覧 マンド設定 Server 8.0\b SQL Server if not exists	型変換一覧 in\mysqldump. r 8.0\bin\mysql

⑥ 確認後、アプリケーションを終了します。

### 3.9. 削除機能の作成

レコードを削除するときは、以下の DELETE 定型文を用意し、「KOBAICD」を削除対象の注文番号で置き 換えます。<sup>14</sup>

delete from irai where kobaiCD='KOBAICD'

#### 追加するコンポーネント

- ラベル
- ボタン(「ボタン(参照)」コンポーネントをコピーして使用)

#### 追加する機能

- 1. [削除]ボタンをクリックすると、
- 2. 文字列格納変数コンポーネントに DELETE 定型文を設定し、
- 3. 「KOBAICD」を画面表示用テーブルの0行0列のセル値で置き換え、
- 4. その SQL 文をイベント番号 0 で実行する。

## 操作 コンポーネントを追加し、接続しましょう。

- ① 「発注依頼」複合コンポーネントへ移動します。
- ② 必要なコンポーネントを追加します。
   作業領域で右クリック-[コンポーネント追加] [画面構成部品] [テキスト] [ラベル]
   とクリックします。
- ③ [ラベル] コンポーネントに属性情報を設定します。属性情報画面を表示し、
   [text]を「delete from irai where kobaiCD='KOBAICD'」、[ComponentKey]を「DELETE 定型文」
   と設定します。
- ④ 「ボタン(参照)」コンポーネントをコピー&ペーストします。
   ボタンを右クリック-[コンポーネントコピー]し、作業領域で右クリック-[コンポーネントペースト]をクリックします。
- ⑤ ボタン名を変更します。属性情報設定画面を表示し、[text]を「削除」と設定します。

<sup>&</sup>lt;sup>14</sup> 購買管理 14.mzax

⑥ 下の表に示すようにコンポーネントを接続します。

削除ボタンをクリックすると文字列格納変数に DELETE SQL 定型文を設定する❶

接続項目	接続関係
接続元コンポーネント	■ ギタン (ID:20_17 KEV:当時)
(イベント発生コンポーネント)	
発生イベント	アクションイベント
接続先コンポーネント❶	■文字列格納変数(ID:20-12)
起動メソッド	文字列を設定する(イベント発生なし)(String)
<引数>	取得方法:メソッド戻り値
	コンポーネント:ラベル(ID:20-16, KEY:DELETE 定型文)
	メソッド/値:ラベルのテキスト文字列を取得する

画面表示用テーブルの0行0列の値を返す❷

接続先コンポーネント❷	■テーブル (ID:20-10, KEY:画面表示用)
起動メソッド	指定セルの値を取得する(int, int)
<引数0>	説明:指定行
	取得方法:固定值
	メソッド/値:0
<引数1>	説明:指定列
	取得方法:固定值
	メソッド/値:0

KOBAICD を画面表示用テーブルの0行0列の値に置き換える❸

接続先コンポーネント€	■文字列格納変数(ID:20-12)
起動メソッド	正規表現に適合する最初の文字列を置換する
	(String, String)
<引数0>	取得方法:固定值
	メソッド/値:KOBAICD
<引数1>	取得方法:メソッド処理結果
	メソッド/値:指定セルの値を取得する

SQL 文をイベント番号0で実行する4

接続先コンポーネント❹	■イベント生成 (ID:20-3, KEY:送信イベント)
起動メソッド	データ生成イベントの発生(Object, int)
<引数0>	説明:生成データ
	取得方法:メソッド戻り値
	コンポーネント : 文字列格納変数 (ID:20-12)
	メソッド/値:文字列を取得する
<引数1>	説明:イベント番号
	取得方法:固定值
	メソッド/値:0

確認 接続関係と配置は以下のようになります。

<ul> <li>         ・・・・・・・・・・・・・・・・・・・・・・・・・・・・・</li></ul>			
■ ラベル ID: 20-16 レビン・20-16			
■ボタン 10:20-17.	アクションイベント	────────────────────────────────────	■ 文字列格納変数 ID: 20-12 TUD: 20-12
(KEY∷"∄IJØ#")		指定セルの値を取得する	KEY: "文字列格納変数12" 「テーブル ID: 20-10
		正規表現に適合する最初の文字列を置換する	IXEY: "面面表示用"           IXEY: "面面表示用"           IXEY: "面面表示用"
		イベント番号:0	ID:20-12 KEY: "文字列格納変数12"
		データ生成イベントの発生	■ 1イベント生成 ID:20-3 KEY: "送信イベント"

図 67 削除機能の作成(コンポーネント接続関係)

## 操作 画面を作成しましょう。

- ① 画面編集ボタンをクリックします。
- ② 画面編集画面で「ボタン」コンポーネントをフレームに追加します。



図 68 削除機能の作成(画面配置)

③ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

## 操作 アプリケーションの動作を確認しましょう。

- ① アプリケーションビルダーのトップ画面に移動します。
- ② 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ③ [接続]ボタンをクリックし、「kobai」データベースに接続します。
- ④ メニューから、[購買管理] [発注依頼…] とクリックします。
- ⑤ [削除]ボタンをクリックします。

§ 発注依頼	<b>A</b>						
					参照	登録	削除
注文番号	注文日	発注部門コード	購買品コード	発注数量	単価	金額	部門承認
A1	2011/11/17	S1	P1	10	10,000	100,000	(F)

図 69 削除機能の動作確認1

 ⑥ メインウィンドウで、「select \* from irai」という SQL 文を実行し、レコードが削除されている ことを確認します。

データベース 購買	買管理								^
select * from irai	)			SQL文実	íτ			複要	竹追加
shonin kit	boKensa s	eihinCD	kiboKobai	. kiboKing	aku kibo	Tanka	kiboSur	yo kibo	Nouki
- テーブル情報 - テーブル名称 [ - テーブル構成 ]	71	ールド名 (道	英数字)	作成 デ <sup>ベ</sup>	更新 - 夕型	削除	 +~		î
	1 175800	1 行前明统	) (V+F			ρī	IJ→覧	型交換	→覧
-ローカルMySQL	1 行法 <u>组加</u> 設定	1 行首明统	 データ/	、 ベース作成	· ⊐ピ- ·	クエ 操作コマ	リー <b>覧</b> 7ンド設定	型交換 :	→覧
ーーカルMySQL」 ドライバ	1 行派的0 設定 	1 行首明独 cj.jdbc.Drive	- ソート データ・ 「 タン	・ ベース作成 ノブ生成	・コピー・ iMySQLW	クエ 操作コマ MySQL S	リー覧 ?ンド設定 erver 8.0\	型交換 ? bin\mysqlo	一覧 Jump.ex
ーーカルMySQL」 ドライバ サーバ	1 行道加 設定 	1 行削除 cj.jdbc.Drive /localhost/	ソート データ・ I タン Mys	、 ベース作成 ノブ生成 SQL起動	・コピー・ ·MySQLW	クエ 操作コマ AySQL S SQL'My:	リー覧 ?ンド設定 erver 8.0\l	型変換 型変換 E bin\mysqlc	一覧 iump.ex nysql.ex
ローカルMySQL ドライバ サーバ ユーザ名	1 行道加 設定 	1 f7#WB# 2j.jdbc.Drive /localhost/	ソート データ・ イ タン MyS データ・	、 マース作成 シフ生成 SQL短動 ベース作成	・コピー・ :WySQL\\ 1 Files\Wy create da	クエ 操作コマ MySQL S SQL'MyS tabase i	リー覧 マンド設定 SQL Serve fnot exist	型交換 空交換 さ bin/mysqld er 8.0\bin/rr	→覧 dump.ex nysql.ex
ローカルMySQL ドライバ サーバ ユーザ名 パスワード	1 行法自加 設定 	1行副除 3.jdbc.Drive /localhost/	ソート データン イ ダン データ・ データ・ データ・	、 ベース作成 シブ生成 SaL超動 ベース作成 ベース指定	・コピー・ :MySQLW 1 FilesWy create da	クエ 操作コマ MySQL S SQL\My: tabase i	リー覧 7ンド設定 erver 8.0\l SQL Serve f not exist	型変換 E bin\mysqlc er 8.0\bin\r s	→覧 Jump.ex nysql.ex
ローカルMySQL ドライバ サーバ ユーザ名 バスワード データベース名	1 行派加 設定 jdbc:mysql.c jdbc:mysql./ root kobai	1 行動除 j.jdbc.Drive /localhost/	ソート データ・ ダン データ・ データ・	・ マース作成 シフ生成 SGL短動 ベース作成 ベース指定	・コピー・ :WySQLW 1 FilesWy create da use	クエ 操作コマ MySQL S SQL MyS tabase i	リー覧 マンド設定 erver 8.0\/ SQL Serve f not exist	型変換 型変換 bin\mysqld er 8.0\bin\r s	一覧 Jump.ex nysql.ex

図 70 削除機能の動作確認 2

⑦ 確認後、アプリケーションを終了します。

## 3.10. 更新機能の作成(登録機能の修正)

レコードの更新には、UPDATE 文を利用しても構いませんが、ここでは該当するレコードを一度削除し、 あらためて更新後のデータでレコードを登録するようにします。<sup>15</sup>

#### 追加するコンポーネント

• サブルーチン

### 追加する機能

- 1. [登録]ボタンをクリックすると、
- 2. 画面表示用テーブルの0行0列のkobaiCDを持つレコードをiraiテーブルから削除し、
- 3. 画面表示用テーブルのデータを irai テーブルへ登録する。

すなわち、すでに作成した登録機能の最初の部分に削除機能を追加することになります。そのために、 [削除]ボタンに接続している処理をサブルーチンとしてまとめ、そのサブルーチンを[登録]ボタンから 起動するようにします。

# 操作 コンポーネントを追加し、接続しましょう。

- ① 「発注依頼」複合コンポーネントへ移動します。
- ② 必要なコンポーネントを追加します。
   作業領域で右クリック-[コンポーネント追加] [処理部品] [サブルーチン] [サブルーチン] とクリックします。
- ③ コンポーネントキーを設定します。属性情報設定画面を表示し、[ComponentKey]を「削除処理」と 設定します。
- ④ 「サブルーチン(削除処理)」コンポーネントにアクションイベントを追加します。
- ⑤ 「ボタン(削除)」コンポーネントから接続されているメソッドを全てコピーします。 アクションイベント上で右クリック- [起動メソッドコピー] とクリックします。
- ⑥ 「サブルーチン(削除処理)」コンポーネントにメソッドを貼り付けます。
   アクションイベント上で右クリックー [起動メソッド貼り付け] とクリックします。
- ⑦ 「ボタン(削除)」コンポーネントから接続されているメソッドを全て削除します。
   アクションイベント上で右クリックー [イベント処理削除] とクリックします。

<sup>&</sup>lt;sup>15</sup> 購買管理 15.mzax

⑧ 下の表に示すようにコンポーネントを接続します。

削除ボタンをクリックすると削除処理サブルーチンを呼び出す

接続項目	接続関係
接続元コンポーネント	■ ギタン (ID:20_17 KEV:当時)
(イベント発生コンポーネント)s	
発生イベント	アクションイベント
接続先コンポーネント	■サブルーチン(ID:20-18, KEY:削除処理)
起動メソッド	処理を呼び出す()

登録ボタンをクリックすると削除処理サブルーチンを呼び出す

接続項目	接続関係
接続元コンポーネント	■ ギカン (1D:20_0 KEV: 発得)
(イベント発生コンポーネント)	■ バダン(ID:20-9, NEI:豆酥/
発生イベント	アクションイベント
接続先コンポーネント	■サブルーチン(ID:20-18, KEY:削除処理)
起動メソッド	処理を呼び出す()

 「ボタン(登録)」コンポーネントのアクションイベントから接続されている起動メソッドの 順番を入れ替えます。「サブルーチン(削除処理)」メソッドを1番目にします。

## 確認 接続関係と配置は以下のようになります。



図 71 更新機能の作成(登録機能の修正)

- 操作 アプリケーションの動作を確認しましょう。
- アプリケーションビルダーのトップ画面に移動します。
- ② 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ③ [接続]ボタンをクリックし、「kobai」データベースに接続します。
- ④ メニューから、[購買管理] [発注依頼…] とクリックします。
- ⑤ アプリケーションを実行し、更新機能の動作確認を行います。
- ⑥ 確認後、アプリケーションを終了します。

### 3.11. 注文番号一覧設定機能の作成

登録済みの発注依頼を参照するとき、注文番号の一覧が設定されていると便利です。そこで、画面表 示用テーブルの第0列に、登録済み注文番号のリストを設定できるようにします。<sup>16</sup>

#### 追加するコンポーネント

- ラベル
- ボタン
- サブルーチン

### 追加する機能

- 1. [注文番号一覧設定]ボタンをクリックすると、
- 2. 注文番号一覧を取得するための SQL 文をイベント番号 4 で実行し、
- 3. 取得した結果を画面表示用テーブルの第0列の選択リストに設定する。
- 4. 終了時に、設定した選択リストを削除する。

## 操作 コンポーネントを追加し、接続しましょう。

- ① 「発注依頼」複合コンポーネントへ移動します。
- ② 必要なコンポーネントを追加します。
   作業領域で右クリック-[コンポーネント追加] [画面構成部品] [テキスト] [ラベル]
   とクリックします。
   作業領域で右クリック-[コンポーネント追加] [画面構成部品] [ボタン] [ボタン]
   とクリックします。
   作業領域で右クリック-[コンポーネント追加] [処理部品] [サブルーチン] [サブルーチン] とクリックします。
- ③ 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。
   [ラベル] コンポーネントの[text]を「select kobaiCD from irai order by kobaiCD」、
   [ComponentKey]を「注文番号一覧取得」、
   [ボタン] コンポーネントの[text]を「注文番号一覧設定」、
   [サブルーチン]コンポーネントの[ComponentKey]を「注文番号一覧取得」
   と設定します。

<sup>16</sup> 購買管理 16.mzax

下の表に示すようにコンポーネントを接続します。

注文番号一覧設定ボタンをクリックすると注文番号一覧を取得する SQL 文をイベント番号4 で実行する

接続項目	接続関係
接続元コンポーネント	■ ギカン (ID:20_20 KEV:注立来日 監訳字)
(イベント発生コンポーネント)	■ハダン(ID:20-20, NLI:注文田号一見設定)
発生イベント	アクションイベント
接続先コンポーネント	■イベント生成(ID:20-3, KEY:送信イベント)
起動メソッド	データ生成イベントの発生(Object,int)
<引数0>	説明:生成データ
	取得方法:メソッド戻り値
	コンポーネント : ラベル(ID:20-19, KEY:注文番号一覧取得)
	メソッド/値:ラベルのテキスト文字列を取得する
<引数1>	説明:イベント番号
	取得方法:固定值
	メソッド/値:4

イベントを注文番号一覧取得サブルーチンに渡して処理を呼び出す

接続項目	接続関係
接続元コンポーネント	■ノベント生式 (ID:20_2 KEV: 受信ノベント)
(イベント発生コンポーネント)	
発生イベント	データ生成イベント
接続先コンポーネント	■サブルーチン(ID:20-21, KEY:注文番号一覧取得)
起動メソッド	イベントを渡して処理を呼び出す(PFEvent)
<引数>	説明:イベント
	取得方法:イベント
イベント番号	4

データ生成イベントのイベント対象データ(検索結果)をテーブル格納変数に設定する●

接続項目	接続関係
接続元コンポーネント	■サゴルーチン(ID:20_21 KEV:注文来早一覧取得)
(イベント発生コンポーネント)	■ 5 <i>5 ルー 7 5</i> (10:20 <sup>-</sup> 21, NET: 注文田 5 見取付)
発生イベント	データ生成イベント
接続先コンポーネント❶	■テーブル格納変数(ID:20-13)
起動メソッド	テーブルを設定する(PFObjectTable)
<引数>	説明:設定するテーブル
	取得方法:イベント内包
	メソッド/値:イベント対象データ

テーブル格納変数に格納されているデータの第0列をリスト形式で取得する❷

接続先コンポーネント❷	■テーブル格納変数(ID:20-13)
起動メソッド	列データリストを位置指定で取得する(int)
<引数>	取得方法:固定值
	メソッド/値:0

取得した結果を画面表示用テーブルの第0列の選択リストに設定する●

接続先コンポーネント❸	■テーブル (ID:20-10, KEY:画面表示用)
起動メソッド	指定列に選択項目リストを設定する(PFObjectList,int)
<引数0>	説明:選択項目リスト
	取得方法:メソッド処理結果
	メソッド/値:列データリストを位置指定で取得する
<引数1>	説明:指定列
	取得方法:固定值
	メソッド/値:0

終了時に設定した選択リストを削除する

接続項目	接続関係
接続元コンポーネント	■サブル <u>ーチン(ID:20-5</u> KEV: 終了加囲)
(イベント発生コンポーネント)	
発生イベント	アクションイベント
接続先コンポーネント	■テーブル(ID:20-10, KEY:画面表示用)
起動メソッド	指定列に選択項目リストを設定する(PFObjectList, int)
<引数0>	説明:選択項目リスト
	取得方法:固定值
	メソッド/値:なし
<引数1>	説明:指定列
	取得方法:固定值
	メソッド/値:0

⑦ 「サブルーチン(終了処理)」コンポーネントのアクションイベントから接続されている
 起動メソッドの順番を入れ替えます。「テーブル(画面表示用)」のメソッドを上に配置します。



確認 接続関係と配置は以下のようになります。

図 72 注文番号一覧設定機能の作成(コンポーネント接続関係)

## 操作 画面を作成しましょう。

- 画面編集 ボタンをクリックします。
- ② 画面編集画面で「ボタン」コンポーネントをフレームに追加します。



図 73 注文番号一覧設定機能の作成(画面配置)

③ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

## 操作 アプリケーションの動作を確認しましょう。

- ① アプリケーションビルダーのトップ画面に移動します。
- ② 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ③ [接続]ボタンをクリックし、「kobai」データベースに接続します。
- ④ メニューから、[購買管理] [発注依頼…] とクリックします。
- 5 [注文番号一覧設定]ボタンをクリックします。
- ⑥ [注文番号]のセルをクリックし、選択リストが設定されることを確認します。

			参照	登録	削除	注文番号	一覧設定
汪文番号	注文日	発注部門コード	購買品コード	発注数量	単価	金額	部門承認
A1 🚽	2011/11/17	S1	P1	5	10,000	50.000	(m)

図 74 注文番号選択リストの設定

⑦ 確認後、アプリケーションを終了します。
#### 3.12. 金額計算機能の作成

発注数量と単価を入力したとき、自動的に金額を計算する機能を作成します。17

#### 追加するコンポーネント

- イベント生成
- サブルーチン
- 乗算

#### 追加する機能

- 1. 画面表示用テーブルのセルが更新されたとき、
- 2. 画面表示用テーブルの第4列あるいは第5列が選択されていれば、
- 3. 画面表示用テーブルの選択状態を解除し、
- 4. 第0行4列のセル値と第0行5列のセル値の積を第0行6列のセルに設定する。

#### !!!注意事項!!!

乗算実行結果を第0行6列のセルに設定する前に、画面表示用テーブルの選択状態を解除する設定を 忘れると無限ループに陥ります。

操作 コンポーネントを追加し、接続しましょう。

- ① 「発注依頼」複合コンポーネントへ移動します。
- 必要なコンポーネントを追加します。

作業領域で右クリック- [コンポーネント追加] - [処理部品] - [イベント] - [イベント生成] とクリックします。 作業領域で右クリック- [コンポーネント追加] - [処理部品] - [サブルーチン] -[サブルーチン] とクリックします。 作業領域で右クリック- [コンポーネント追加] - [処理部品] - [演算制御] - [乗算(×)] とクリックします。

③ 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。
 [イベント生成] コンポーネントの[ComponentKey]を「セル値更新」、
 [サブルーチン]コンポーネントの[ComponentKey]を「発注数量×単価の計算」と設定します。

<sup>17</sup> 購買管理 17.mzax

下の表に示すようにコンポーネントを接続します。

画面表示用テーブルのセルが更新されたとき更新されたセルの列インデックスを伝達する

接続項目	接続関係
接続元コンポーネント	■ニーゴル (ID:20_10 KEV:両西まテ田)
(イベント発生コンポーネント)	「「) / / / / / / / / / / / / / / / / / /
発生イベント	データ更新イベント
接続先コンポーネント	■イベント生成(ID:20-23, KEY:セル値更新)
起動メソッド	アクションイベントの発生(int)
<引数>	説明:イベント番号
	取得方法:メソッド戻り値
	コンポーネント:テーブル(ID:20-10, KEY:画面表示用)
	メソッド/値:選択列の位置を取得する
イベント番号	0

発注数量×単価の計算サブルーチンを呼び出す

接続項目	接続関係
接続元コンポーネント	■ノベント仕式 (ID:20_22 KEV:セル体再新)
(イベント発生コンポーネント)	■1ハント王成(ID-20-23, NET-セル恒更利)
発生イベント	アクションイベント
接続先コンポーネント	■サブルーチン(ID:20-24, KEY:発注数量×単価の計算)
起動メソッド	処理を呼び出す()
イベント番号	4、5

画面表示用テーブルの選択状態を解除する●

接続項目	接続関係
接続元コンポーネント	■サゴルーチン(ID:20_24 KEV:発汁粉号×単価の計質)
(イベント発生コンポーネント)	
発生イベント	アクションイベント
接続先コンポーネント❶	■テーブル (ID:20-10, KEY:画面表示用)
起動メソッド	clearSelection()

第0行4列のセル値を取得する2

接続先コンポーネント❷	■テーブル (ID:20-10, KEY:画面表示用)
起動メソッド	指定セルの値を取得する(int, int)
<引数0>	説明:指定行
	取得方法:固定值
	メソッド/値:0
<引数1>	説明:指定列
	取得方法:固定值
	メソッド/値:4

第0行5列のセル値を取得する❸

接続先コンポーネント€	■テーブル (ID:20-10, KEY:画面表示用)
起動メソッド	指定セルの値を取得する(int, int)
<引数0>	説明:指定行
	取得方法:固定值
	メソッド/値:0
<引数1>	説明:指定列
	取得方法:固定值
	メソッド/値:5

発注数量(第0行4列)×単価(第0行5列)を計算する④

接続先コンポーネント❹	■乗算(×)(ID:20-22)
起動メソッド	数値変換/左右オペランド設定後、演算を行う
	(String, String)
<引数0>	説明:左オペランド
	取得方法:メソッド処理結果
	メソッド/値:指定セルの値を取得する(第0行4列)
<引数1>	説明:右オペランド
	取得方法:メソッド処理結果
	メソッド/値:指定セルの値を取得する(第0行5列)

乗算実行結果を第0行6列のセルに設定する

接続項目	接続関係
接続元コンポーネント	■ 垂笛 ( ~ ) (ID:20_22)
(イベント発生コンポーネント)	■米异(^) (ID·20 <sup>-</sup> 22)
発生イベント	処理完了イベント
接続先コンポーネント	■テーブル (ID:20-10, KEY:画面表示用)
起動メソッド	セルの値を設定する(Object, int, int)
<引数0>	説明:セルの値
	取得方法:イベント内包
	メソッド/値:処理結果データ
<引数1>	説明:行の位置
	取得方法:固定值
	メソッド/値:0
<引数2>	説明:列の位置
	取得方法:固定值
	メソッド/値:6
イベント番号	0

#### 確認 接続関係と配置は以下のようになります。

		データ更新イベントの発生	<mark>  </mark> イベント生成   D:20-3
			KEY:"送信イベント"
□ テーブル	データ更新イベント	アクションイベントの発生	●イベント生成
ID:20-10 KEY:"画面表示用"		[NO:0]	ID:20-23 KEY:"セル値更新"
■イベント生成	アクションイベント	処理を呼び出す	<mark> </mark> サブルーチン
ID : 20-23 KEY : "セル値更新"		[N0:4,5]	D:20-24  KEY:"発注数量×単価の計算"
☐ サブルーチン	アクションイベント	clearSelection	<u> </u> テーブル
ID:20-24 KEY:"発注数量×単価の計算"			ID:20-10 KEY:"画面表示用"
		第0行4列のセル値取得	
		指定セルの値を取得する	<u> </u>
			D:20-10  KEY:"画面表示用"
		第0行5列のセル値取得	
		指定セルの値を取得する	■ テーブル
			ID:20-10 KEY:"画面表示用"
		   数値変換/左右オペランド設定後、演算を行う	<mark>  </mark> 乗算(×)
			ID : 20-22 KEY : "乗算(×)22"
● 乗算(×)	処理完了イベント	セルの値を設定する	<u> </u>
ID : 20-22 KEY : "兼算(×)22"		[NO:0]	ID:20-10 KEY:"画面表示用"
ID:20-11 KEY:"登録用"			

#### 図 75 金額計算機能の作成

# 操作 アプリケーションの動作を確認しましょう。

- ① アプリケーションビルダーのトップ画面に移動します。
- ② 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ③ [接続]ボタンをクリックし、「kobai」データベースに接続します。
- ④ メニューから、[購買管理] [発注依頼…] とクリックします。
- ⑤ 金額計算機能の動作確認を行います。
- ⑥ 確認後、アプリケーションを終了します。

## 3.13. 伝票発行機能の作成

下図のような伝票を発行する機能を作成します。18



図 76 伝票発行

#### 追加するコンポーネント

- ラベル(注文番号用)
- ラベル(発注部門コード用)
- ラベル(購買品コード用)
- ラベル(単価用)
- ラベル(発注数量用)
- 実数(Double)格納変数(金額用)
- ボタン
- サブルーチン(帳票用ラベル設定用)
- サブルーチン(帳票用テーブル作成用)
- オブジェクトキュー
- 帳票

#### 追加する機能

- 1. [伝票発行…]ボタンをクリックすると、
- 2. 画面表示用テーブルのセルの値をラベルおよび実数(Double)格納変数コンポーネントに設定 するとともに、
- 3. 印刷用のテーブル(注文日と部門承認欄用)を作成し、
- 4. それらのデータに基づく発注依頼書の印刷プレビューを表示する。

<sup>18</sup> 購買管理 18.mzax

#### 操作 コンポーネントを追加し、接続しましょう。

- ① 「発注依頼」複合コンポーネントへ移動します。
- ② 必要なコンポーネントを一括追加します。
   作業領域で右クリック-[コンポーネントー括追加...]とクリックします。
   [画面構成部品]パネルが表示されていることを確認します。
   右側の領域から[テキスト]をクリックし、[ラベル]の追加数に「5」と入力します。
   右側の領域から[ボタン]をクリックし、[ボタン]の追加数に「1」と入力します。
   [処理部品]パネルに切り替えます。
   右側の領域から[変数]をクリックし、[浮動小数点数(Double)格納変数]の追加数に「1」と
   入力します。
   右側の領域から[世ブルーチン]をクリックし、[サブルーチン]の追加数に「2」と入力します。
   右側の領域から[配列]をクリックし、[オブジェクトキュー]の追加数に「1」と入力します。
   [入出力]パネルに切り替えます。
   右側の領域から[帳票]をクリックし、[帳票]の追加数に「1」と入力します。
   最後に[追加]ボタンをクリックします。
- 3 属性情報を設定します。属性情報画面を表示し、それぞれ次のように設定します。
   注文番号用として追加した[ラベル] コンポーネントの[ComponentKey]を「注文番号」、
   発注部門コード用として追加した[ラベル] コンポーネントの[ComponentKey]を「発注部門コード」、
   購買品コード用として追加した[ラベル] コンポーネントの[ComponentKey]を「購買品コード」、
   単価用として追加した[ラベル] コンポーネントの[ComponentKey]を「単価」、
   発注数量用として追加した[ラベル] コンポーネントの[ComponentKey]を「発注数量」、
   [ボタン] コンポーネントの[text]を「伝票発行…」、
   帳票用ラベル設定用として追加した[サブルーチン]コンポーネントの[ComponentKey]を

「帳票用ラベル設定」、

- 帳票用テーブル作成用として追加した[サブルーチン]コンポーネントの[ComponentKey]を 「帳票用テーブル作成」
- と設定します。

下の表に示すようにコンポーネントを接続します。

伝票発行ボタンをクリックすると帳票用ラベル設定サブルーチンを呼び出す●

接続項目	接続関係
接続元コンポーネント	■ ギカヽ, (ID:20_21 KEV: /二 西 み /二 )
(イベント発生コンポーネント)	■ バダン(ID-20-31, NEI・伝示光1])
発生イベント	アクションイベント
接続先コンポーネント❶	■サブルーチン(ID:20-33, KEY:帳票用ラベル設定)
起動メソッド	処理を呼び出す()

伝票発行ボタンをクリックすると帳票用テーブル作成サブルーチンを呼び出す❷

接続先コンポーネント❷	■サブルーチン(ID:20-34, KEY:帳票用テーブル作成)
起動メソッド	処理を呼び出す()

伝票発行ボタンをクリックすると印刷プレビュー画面を表示する❸

接続先コンポーネント€	■帳票 (ID:20-35)
起動メソッド	印刷プレビュー画面を表示する(Component)
<引数>	説明:親コンポーネント
	取得方法:固定值
	メソッド/値:なし

画面表示用テーブルの第0行のデータを取得する●

接続項目	接続関係
接続元コンポーネント	■サゴル―チン(ID:20_22 KEV:帳専田ラズル設字)
(イベント発生コンポーネント)	■ 9 クルー 7 ク(ID-20-33, ILT: 帳票用 グベル設定)
発生イベント	アクションイベント
接続先コンポーネント❶	■テーブル(ID:20-10, KEY:画面表示用)
起動メソッド	指定行の行データをリスト形式で取得する(int)
<引数>	取得方法:固定值
	メソッド/値:0

取得したデータをキューとして設定する❷

接続先コンポーネント❷	■オブジェクトキュー (ID:20-32)
起動メソッド	リストによるキューの設定(PFObjectList)
<引数>	取得方法:メソッド処理結果
	メソッド/値:指定行の行データをリスト形式で取得する

ラベルに注文番号を設定する₿

接続先コンポーネント❸	■ラベル(ID:20-25, KEY:注文番号)
起動メソッド	ラベルのテキスト文字列を設定する(String)
<引数>	説明:テキスト文字列
	取得方法:メソッド戻り値
	コンポーネント : オブジェクトキュー(ID:20-32)
	メソッド/値:オブジェクトの取得

オブジェクトを取得する④

接続先コンポーネント❹	■オブジェクトキュー (ID:20-32)
起動メソッド	オブジェクトの取得()

ラベルに発注部門コードを設定する€

接続先コンポーネント❺	■ラベル(ID:20-26, KEY:発注部門コード)
起動メソッド	ラベルのテキスト文字列を設定する(String)
<引数>	説明:テキスト文字列
	取得方法:メソッド戻り値
	コンポーネント:オブジェクトキュー(ID:20-32)
	メソッド/値:オブジェクトの取得

## ラベルに購買品コードを設定する❻

接続先コンポーネント❻	■ラベル(ID:20-27, KEY:購買品コード)
起動メソッド	ラベルのテキスト文字列を設定する(String)
<引数>	説明:テキスト文字列
	取得方法:メソッド戻り値
	コンポーネント:オブジェクトキュー(ID:20-32)
	メソッド/値:オブジェクトの取得

ラベルに発注数量を設定する♥

接続先コンポーネント❼	■ラベル(ID:20-28, KEY:発注数量)
起動メソッド	ラベルのテキスト文字列を設定する(String)
<引数>	説明:テキスト文字列
	取得方法:メソッド戻り値
	コンポーネント:オブジェクトキュー(ID:20-32)
	メソッド/値:オブジェクトの取得

ラベルに単価を設定する**⑧** 

接続先コンポーネント❸	■ラベル(ID:20-29, KEY:単価)
起動メソッド	ラベルのテキスト文字列を設定する(String)
<引数>	説明:テキスト文字列
	取得方法:メソッド戻り値
	コンポーネント : オブジェクトキュー(ID:20-32)
	メソッド/値:オブジェクトの取得

ラベルに金額を設定する᠑

接続先コンポーネント❷	■実数(Double)格納変数 (ID:20-36, KEY:金額)
起動メソッド	数値(Double)を設定する(Double)
<引数>	説明:設定する Double オブジェクト
	取得方法:メソッド戻り値
	コンポーネント:オブジェクトキュー(ID:20-32)
	メソッド/値:オブジェクトの取得

画面表示用テーブルデータをテーブル格納変数に設定する●

接続項目	接続関係
接続元コンポーネント	■サゴリ、(ID:20_24 KEV:帳亜田ニゴル佐式)
(イベント発生コンポーネント)	■ リノルーテン(ID:20-34, NEI:帳奈用) - ノル(F成)
発生イベント	アクションイベント
接続先コンポーネント❶	■テーブル格納変数(ID:20-13)
起動メソッド	テーブルを設定する(PFObjectTable)
<引数>	説明:設定するテーブル
	取得方法:メソッド戻り値
	コンポーネント : テーブル(ID:20-10, KEY:画面表示用)
	メソッド/値:テーブルデータを取得する

第6列を削除する2

接続先コンポーネント❷	■テーブル格納変数(ID:20-13)
起動メソッド	列を位置指定で削除する(int)
<引数>	説明:指定位置
	取得方法:固定值
	メソッド/値:6

## 第5列を削除する❸

接続先コンポーネント€	■テーブル格納変数(ID:20-13)
起動メソッド	列を位置指定で削除する(int)
<引数>	説明:指定位置
	取得方法:固定值
	メソッド/値:5

## 第4列を削除する❹

接続先コンポーネント❹	■テーブル格納変数(ID:20-13)
起動メソッド	列を位置指定で削除する(int)
<引数>	説明:指定位置
	取得方法:固定值
	メソッド/値:4

## 第3列を削除する

接続先コンポーネント❺	■テーブル格納変数(ID:20-13)	
起動メソッド	列を位置指定で削除する(int)	
<引数>	説明:指定位置	
	取得方法:固定值	
	メソッド/値:3	

## 第2列を削除する6

接続先コンポーネント❻	■テーブル格納変数(ID:20-13)
起動メソッド	列を位置指定で削除する(int)
<引数>	説明:指定位置
	取得方法:固定值
	メソッド/値:2

## 第0列を削除する♥

接続先コンポーネント❼	■テーブル格納変数(ID:20-13)
起動メソッド	列を位置指定で削除する(int)
<引数>	説明:指定位置
	取得方法:固定值
	メソッド/値:0

# 確認 接続関係と配置は以下のようになります。

	アクションイベント	ha and also as a line -	(サブルーチン)
ID: 20-31		処理を呼び出す	ID: 20-33
(KEY:"広票発行"			(KEY:"帳票用ラベル設定"
		処理を呼び出す	サフルーチン ID:20:34
			KEY:"帳票用テーブル作成"
		印刷プレビュー画面を表示する	
			ID:20-35 KEY:"帳覃35"
サブルーチン	アクションイベント		( テーブル
ID: 20-33		指定行の行ナータをリスト形式で取得する	ID: 20-10
(KEY:"帳票用ラベル設定"]			KEY:"画面表示用"
		リストによるキューの設定	D:20-32
			KEY:"オブジェクトキュー32"
		ラベルのテキスト文字列を設定する	
			ID:20-25 KEY:"注文番号"
		オゴジェクトの取得	()オブジェクトキュー
		7.7717104XH	ID: 20-32
		ラベルのテキスト文字列を設定する	1D : 20-26
			KEY:"発注部門コード"
		ラベルのテキスト文字列を設定する	
			ID:20-27 KEY:"購買品コード"
		ラベルのテキスト文字列を設定する	■ ラベル
		J THE THE ATTICALLY D	ID:20-28 KFY:"從注約量"
		ラベルのテキスト文字列を設定する	ID: 20-29
			KEY:"単価"
		数値(Double)を設定する	■ 実数(Double)格納変数
			ID.20-36 KEY:"金額"
マー・100-000-1           ラベル           D: 20-38           KEY:*単価           リラベル           D: 20-29           KEY:*単価           「大学、小会語           「オブジェクトキュー           [D: 20-36           「オブジェクトキュー			
KEY:"オブジェクトキュー32"			
UD:20-34	回田表示用ナーフルナータをナーブル格	MY1変要別に設定し、弗6、5、4、3、2、U外で削除」	「テーブル」検幼亦断
KEY:"帳票用テーブル作成"		テーブルを設定する	ID: 20-13
			(KEY:"テーブル格納変数13"
		列を位置指定で削除する	
			KEY:"テーブル格納変数13"
		列を位置指定で削除する	テーブル格納変数
			ID: 20-13 KEY: "テーブル格納変数13"
		加大/古里北宁本約80余子ェ	<ul> <li>テーブル格納変数</li> </ul>
		クリン 山山直相足 で用即本 9 つ	ID:20-13 レビン・"テーーブル」検知が考れつ"
		列を位置指定で削除する	ID: 20-13
			KEY:"テーブル格納変数13"
		列を位置指定で削除する	
			ID: 20-13 KEY:"テーブル格納変数13"
		列を位置指定で削除する	
		/ ICHLEIBAC CRIPE? &	ID:20-13 KEY: "テーブル格納変数13"
「「「「「「「」」」			CALL & STOTEM BERKING
ID:20-35 KEY:"帳票35"			

## 図 77 伝票発行機能の作成(コンポーネント接続関係)

## 操作 | 画面を作成しましょう。

- ① 画面編集 ボタンをクリックします。
- ② 画面編集画面で「ボタン」コンポーネントをフレームに追加します。

編集				
□	フレーム (ID:20-1) (KEY:"発注依頼")			
————————————————————————————————————	参照 登録 自即来 汪文書号一覧設定 伝票発行…			
<ul> <li>ボタン [ID:20-17] (KEY:消除)</li> <li>ボタン (ID:20-20) (KEY:消除)</li> </ul>	注文番号 注文日 発注部門コード 購買品コード 発注数量 単価 金額 部門2本認			
「「「「ホタン」[ID:20-20] (KET: /主文番号一覧設定) 「「「ボタン [ID:20-31] (KEY:"伝票発行…")				
5				
	K			
	配置 手動配置 ▼ 図 自動サイズ設定 グリッド間隔5			
	0 10 20 30 40 50 閉じる			

#### 図 78 伝票発行機能の作成(画面配置)

③ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

## 操作 伝票作成に使うデータを設定しておきましょう。

- ① 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ② メニューから、[購買管理] [発注依頼…] とクリックします。
- ③ [伝票発行…]ボタンをクリックします。
   空の[帳票プレビュー]ウィンドウが表示されます。
- ④ [帳票プレビュー] ウィンドウを閉じます。
- ⑤ アプリケーションを終了します。

## 操作 印刷用の画面を作成しましょう。

- 帳票編集 ボタンをクリックします。
- (2) 帳票サイズと帳票方向を設定します。
   [用紙設定...] ボタンをクリックし、[A5 サイズ]、[横]に設定します。

用紙サイス	(	
A5サ-	( <u>z</u> ]•] 🛛 !	従 📝 横
幅	210 🕀 高さ	148 ≑
余白		
	<u> </u>	
左 20		右 20 ≑
	下 20美	
	hanned.	
		+7112

#### 図 79 伝票発行機能の作成(用紙設定)

- ③ タイトル用のラベル要素をレイアウト画面に追加します。
   レイアウト画面上で右クリック- [帳票要素追加] [ラベル要素] とクリックします。
- ④ データを設定します。[ラベル要素]上で右クリック- [データ設定] [テキスト入力…] と クリックします。「発注依頼書」と入力します。
   適当な場所へ移動し、大きさを調整します。
- ⑤ テーブル要素をレイアウト画面に追加します。
   レイアウト画面上で右クリックー [帳票要素追加] [テーブル要素] とクリックします。
- ⑥ データを設定します。[テーブル要素]上で右クリック- [データ設定] [データ取得設定...]
   とクリックします。[取得元コンポーネント]は[テーブル格納変数]、[データ取得メソッド]は
   [テーブルを取得する()]を選択します。
   適当な場所へ移動し、大きさを調整します。



#### 図 80 伝票発行機能の作成(画面配置1)

#### ワンポイント 一描画属性設定一

ラベル要素の文字フォントや文字色、文字表示位置、罫線色などの属性は、[ラベル描画設定] 画面 で変更できます。

① レイアウト画面の [ラベル要素] 上で右クリックー [ラベル描画属性設定…] をクリックします。

文字フォント	変更	プレビュー
文字色	変更	<b>重</b> あぁアァアオ Sample 123
下線	🔲 表示する	and a state of the
縦余白(Point)	5 *	
横余白(Point)	5	
行間隔(Point)	0	
文字位置	[左上 ▼]	
背景色	変更	
罫線色	変更	
罫線太さ (Point)	0.5	
📃 繰り返し印刷		
		「設定」「キャンセル」

② 任意の設定後設定ボタンをクリックします。

⑦ ラベル要素をレイアウト画面に12個追加し、下の表に示すようにデータを設定します。

番号	データ設定	入力内容	取得元コンポーネント	データ取得メソッド
1	テキスト入力	注文番号		
2	テキスト入力	発注部門コード		
3	テキスト入力	購買品コード		
4	データ取得設定		ラベル(注文番号)	ラベルのテキスト文字列を取
				得する()
5	データ取得設定		ラベル(発注部門コー	ラベルのテキスト文字列を取
			ド)	得する()
6	データ取得設定		ラベル(購買品コード)	ラベルのテキスト文字列を取
				得する()
7	テキスト入力	発注数量		
8	テキスト入力	単価		
9	テキスト入力	金額		
10	データ取得設定		ラベル(発注数量)	ラベルのテキスト文字列を取
				得する()
11	データ取得設定		ラベル(単価)	ラベルのテキスト文字列を取
				得する()
12	データ取得設定		実数(Double)格納変数	long 値を取得する()
			(金額)	



図 81 伝票発行機能の作成(画面配置 2)

⑧ 閉じるボタンをクリックして、アプリケーションビルダー画面に戻ります。

## 操作 アプリケーションの動作を確認しましょう。

- ① アプリケーションビルダーのトップ画面に移動します。
- ② 実行もしくは実行(設定可)ボタンをクリックし、アプリケーションを起動します。
- ③ [接続]ボタンをクリックし、「kobai」データベースに接続します。
- ④ メニューから、[購買管理] [発注依頼…] とクリックします。
- ⑤ [伝票発行…] ボタンをクリックします。
- ⑥ 確認後、アプリケーションを終了します。

## 4. コンポーネント接続関係(まとめ)

トップ階層



<mark>-1 メニュー</mark> 10:8			
(KEY:"データベース")	アクションイベント		
D:9  FV:"」ズックアッチ "		単数Save用ファイル選択ダイアログを表示する	5000000000000000000000000000000000000
「ファイル選択	データ選択イベント	cotFile	「テキストファイル出力
ID:10 KEY:"ダンプ出力ファイル選択"		- Servie	[NO:1] ID:11 KEY:"ダンプファイル出力"
		ダンプ生成	「ローカルMySQL管理」
			[NO:1] ID:2 KEY:"ローカルMySQL管理"
ラキストファイル出力			
ID:11 KEY:"ダンブファイル出力"			
	アクションイベント	単数Open用ファイル選択ダイアログを表示する	5 ファイル選択
KEY:"内容設定"			KEY:"バックアップファイル選択"
■ ファイル選択 ID:13	データ選択イベント	ファイルオブジェクトを指定して読み込む	<u>テキストファイル入力</u> ID:14
(KEY:")バックアップファイル選択"			[NO.1] [KEY: "バックアップファイル入力"]
		ファイルを閉じる	<b>リテキストファイル人力</b> [No:11] ID:14
	ニーク生命イベント		
$\frac{10.14}{10.14}$		ダンプ実行	
			(KET. H-M/KW/SQLEIE
ID:3 KEY:"SQL *"			
	アクションイベント	イベント番号:1	
ID:4 KEY:"SQL文実行"	•	イベント番号を指定してSQL文を実行する	<u> ローカルMySQL管理</u> D:2
			KEY:"ローカルMySQL管理"
<u> テーブル</u> ID:5			
KEY:"テーブル5"			
(KEY:"道加先テーブル名"	アクションイベント		
10:7 レロン・"対策地が示い自由の"		指定したテーブルに複数行を一括追加する	
			UCL. H = NINMIOGERIE





サブルーチン	アクションイベント	指定行の行データをリスト形式で取得する	
ID:20-33  KFY:"帳専用ラベル語完"			ID:20-10 KEY:"画面表示用"
(Ser. Raning Stream			
		リストによるキューの設定	
			KEY: "オブジェクトキュー32"
		ニベルのテナフト立字列を設定する	[]ラベル
		J. (100) + (1 X + )(2 ax E y 5	ID: 20-25
			KEY:"注义番亏"
		オブジェクトの取得	<u> </u>
			D:20-32  KEY:"オブジェクトキュー32"
		マベルのナキ人ト文子列を設定する	ID: 20-26
			<u>(KEY:"発注部門コード"</u>
		ラベルのテキスト文字列を設定する	[]ラベル
			ID:20-27 KEY:"購買品コード"
		ラベルのテキスト文子列を設定する	ID: 20-28
			(KEY:"発注数量"
		ラベルのテキスト文字列を設定する	
			ID:20-29 KEY:"単価"
		<u>数値(Double)を設定する</u>	ID: 20-36
			KEY:"金額"
<ul> <li>LKEY: ※2:数量"</li> <li>ラベル</li> <li>D: 20-29</li> <li>KEY: "単価"</li> <li>実数(Double)格納変数</li> <li>D: 20-36</li> <li>KEY: *金額"</li> <li>オブジェクトキュー</li> <li>D: 20-32</li> <li>KEY: *オブジェクトキュー</li> </ul>			
[] + JU- FV	画面表示用テーブルデータをテー:	ブル格納変数に設定し、第6、5、4、3、2、0列を削除	
	アクションイベント	ニ テーブルを設定する	ブル格納変数
(ペロン 喉奈田ナーフルギわな")	-		ID:20-13 KFY:"テーブル格納変類13"
		列を位置指定で削除する	ID: 20-13
			【KEY:"テーブル格納変数13"
		列を位置指定で削除する	🛑 テーブル格納変数
		/ TCHERRAC CHIRACY &	ID:20-13 レビン・"テーゴル旅纳変歩12"
		列を位置指定で削除する	<ul> <li></li></ul>
			ID. 20-13 KEY:"テーブル格納変数13"
		だけおけた思わらったいのなった。	<ul> <li>デーブル格納変数</li> </ul>
		2月で12直指正で削除する	ID: 20-13
			KEY:"テーフル格納変数13"
		列を位置指定で削除する	■ テーブル格納変数
			ID:20-13 KEY:"テーブル格納変数13"
			<ul> <li>デーブル格納変数</li> </ul>
		別を位置指定で削除する	ID: 20-13
			LKEY:"テーブル格納変数13"
■ 帳票 ID: 20-35 VEV: "帳==25"			