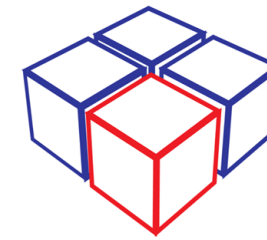


capancioni.com



lema
Software Architectures

Oracle APEX Developing Database Web Applications



6 Aprile 2023

ORACLE

Partner



capancioni.com

Chi Siamo

Ing. Roberto Capancioni

—Sviluppo Oracle APEX



—Board Member ITOUG



Email: sviluppo@capancioni.com

Linkedin: <https://www.linkedin.com/in/robertocapancioni>

ITOUg: <https://itoug.it>

Chi Siamo

- Dott.ssa Nicoletta Bindi
- in Tema dal 2001
- Analista sviluppatrice java e web
- Apex dal 2020

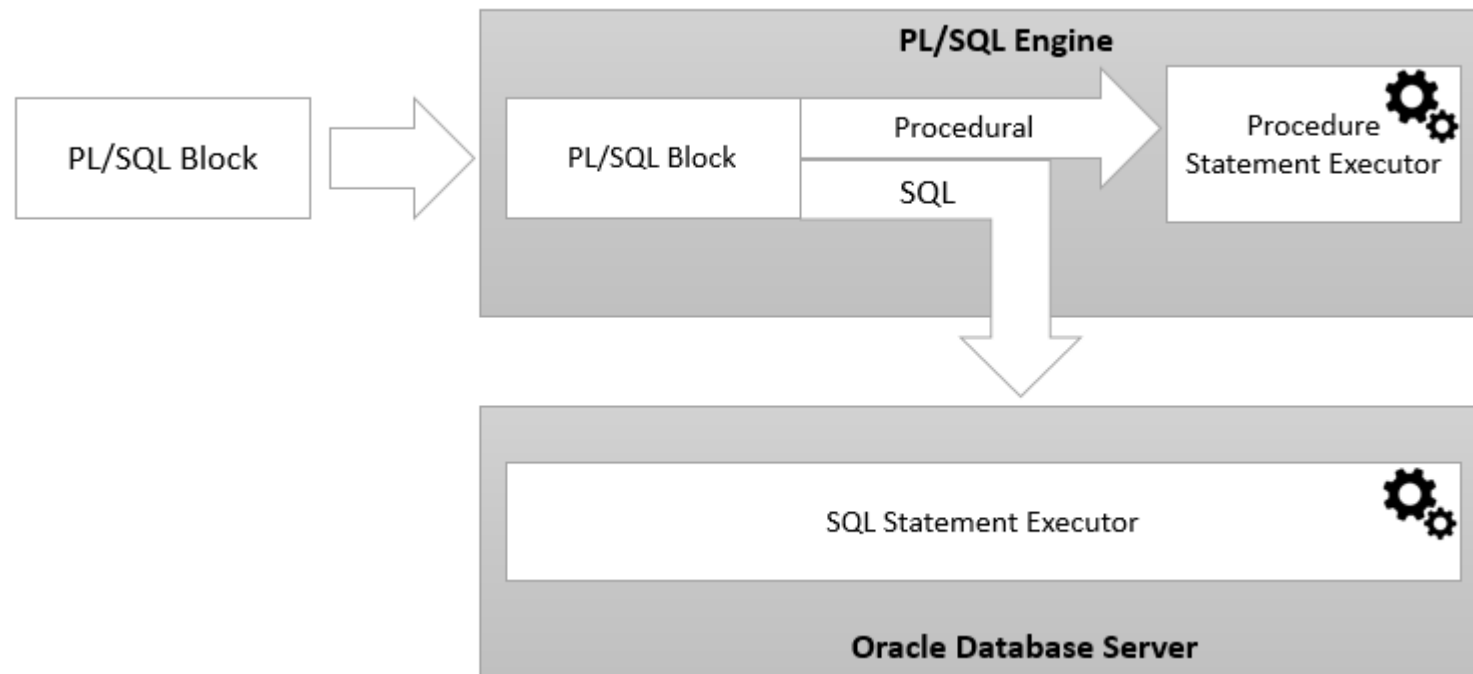


PL/SQL

PL/SQL

- Aggiunge costrutti procedurali al linguaggio SQL
- E' un linguaggio altamente strutturato
- E' integrato all'interno del database Oracle

Procedural
Language extensions to the
Structured
Query
Language



PL/SQL

```
BEGIN
    DBMS_OUTPUT.put_line ('Hello World! ');
END ;
/
```

PL/SQL

DECLARE

 l_message VARCHAR2(255) := 'Hello World!';

BEGIN

 DBMS_OUTPUT.PUT_LINE(l_message);

END;

/

PL/SQL

DECLARE

V_RESULT NUMBER;

BEGIN

V_RESULT := 1 / 0;

EXCEPTION

WHEN ZERO_DIVIDE THEN

DBMS_OUTPUT.PUT_LINE('Non puoi dividere per zero');

WHEN OTHERS THEN

DBMS_OUTPUT.PUT_LINE(**SQLERRM);**

END;

/

Exception	Oracle Error
ACCESS_INTO_NULL	ORA-06530
CASE_NOT_FOUND	ORA-06592
COLLECTION_IS_NULL	ORA-06531
CURSOR_ALREADY_OPEN	ORA-06511
DUP_VAL_ON_INDEX	ORA-00001
INVALID_CURSOR	ORA-01001
INVALID_NUMBER	ORA-01722
LOGIN_DENIED	ORA-01017
NO_DATA_FOUND	ORA-01403
NOT_LOGGED_ON	ORA-01012
PROGRAM_ERROR	ORA-06501
ROWTYPE_MISMATCH	ORA-06504
SELF_IS_NULL	ORA-30625
STORAGE_ERROR	ORA-06500
SUBSCRIPT_BEYOND_COUNT	ORA-06533
SUBSCRIPT_OUTSIDE_LIMIT	ORA-06532
SYS_INVALID_ROWID	ORA-01410
TIMEOUT_ON_RESOURCE	ORA-00051
TOO_MANY_ROWS	ORA-01422
VALUE_ERROR	ORA-06502
ZERO_DIVIDE	ORA-01476

PL/SQL

DECLARE

 l_data date;

BEGIN

SELECT sysdate

INTO l_data

FROM dual;

 DBMS_OUTPUT.PUT_LINE(to_char(l_data, 'DD/MM/YYYY HH24:MM:SS'));

END;

/

PL/SQL

DECLARE

 l_data date;

BEGIN

SELECT sysdate

INTO l_data

FROM dual;

 DBMS_OUTPUT.PUT_LINE(to_char(l_data, 'DD/MM/YYYY HH24:MM:SS'));

END;

/

PL/SQL

DECLARE

l_venduto NUMBER := 100000;

BEGIN

IF l_venduto > 100000 **THEN**

DBMS_OUTPUT.PUT_LINE('Venduto > 100k');

ELSIF l_venduto > 50000 **THEN**

DBMS_OUTPUT.PUT_LINE('Venduto > 50k');

ELSE

DBMS_OUTPUT.PUT_LINE('Venduto <= 50k');

END IF;

END;

/

DECLARE`l_voto CHAR(1);``l_giudizio VARCHAR2(20);`**BEGIN**`l_voto := 'B';`**CASE** `l_voto`**WHEN** `'A'` **THEN**`l_giudizio := 'Eccellente' ;`**WHEN** `'B'` **THEN**`l_giudizio := 'Ottimo' ;`**WHEN** `'C'` **THEN**`l_giudizio := 'Buono' ;`**WHEN** `'D'` **THEN**`l_giudizio := 'Sufficiente' ;`**ELSE**`l_giudizio := 'Insufficiente' ;`**END CASE;**`DBMS_OUTPUT.PUT_LINE(l_giudizio);`**END;****/**

DECLARE

l_venduto NUMBER;

l_commissione NUMBER;

BEGIN

l_venduto := 150000;

CASE**WHEN** l_venduto > 200000 **THEN**

l_commissione := 0.2;

WHEN l_venduto >= 100000 AND l_venduto < 200000 **THEN**

l_commissione := 0.15;

WHEN l_venduto >= 50000 AND l_venduto < 100000 **THEN**

l_commissione := 0.1;

WHEN l_venduto > 30000 **THEN**

l_commissione := 0.05;

ELSE

l_commissione := 0;

END CASE;

DBMS_OUTPUT.PUT_LINE('Commissioni: ' || l_commissione * 100 || '%');

END;

/

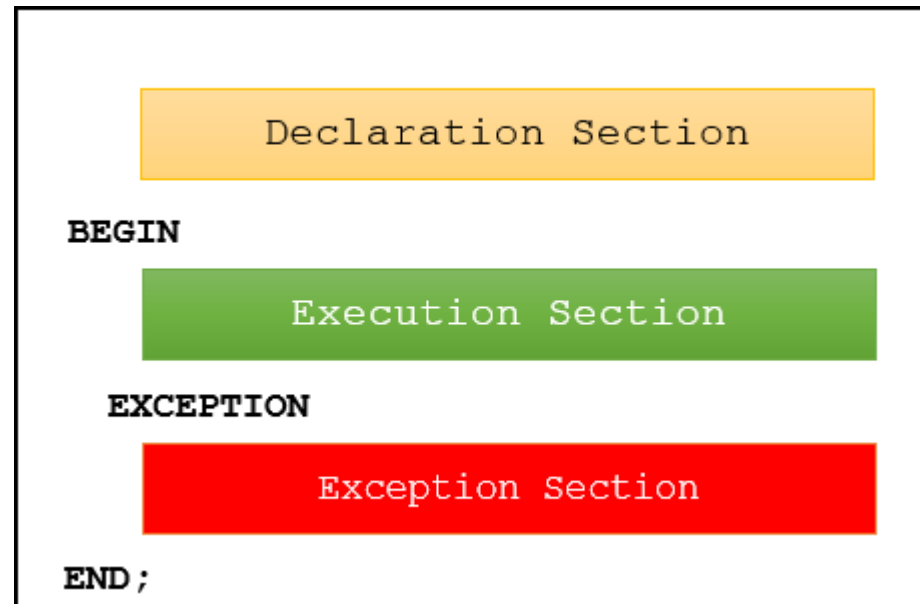
DECLARE

BEGIN

```
    for c in (select level numero from dual connect by level <=10)
    loop
        DBMS_OUTPUT.PUT_LINE( 'Numero--> ' || c.numero );
    end loop;
```

END;

Blocco Anonimo



BLOCCO ANONIMO

```

DECLARE
    l_venduto NUMBER := 100000;
BEGIN
    IF l_venduto > 100000 THEN
        DBMS_OUTPUT.PUT_LINE( 'Venduto > 100k ' );
    ELSIF l_venduto > 50000 THEN
        DBMS_OUTPUT.PUT_LINE( 'Venduto > 50k ' );
    ELSE
        DBMS_OUTPUT.PUT_LINE( 'Venduto <= 50k ' );
    END IF;
END;
/

```

PROCEDURA

```

CREATE OR REPLACE PROCEDURE D06_VENDUTO
(
    p_venduto IN number default 0
)
IS
BEGIN
    IF p_venduto > 100000 THEN
        DBMS_OUTPUT.PUT_LINE( 'Venduto > 100k ' );
    ELSIF p_venduto > 50000 THEN
        DBMS_OUTPUT.PUT_LINE( 'Venduto > 50k ' );
    ELSE
        DBMS_OUTPUT.PUT_LINE( 'Venduto <= 50k ' );
    END IF;
END;
/

BEGIN
    D06_VENDUTO(p_venduto => 100000);
END;
/

```


BLOCCO ANONIMO

```

DECLARE
    l_venduto NUMBER := 100000;
BEGIN
    IF l_venduto > 100000 THEN
        DBMS_OUTPUT.PUT_LINE( 'Venduto > 100k ' );
    ELSIF l_venduto > 50000 THEN
        DBMS_OUTPUT.PUT_LINE( 'Venduto > 50k ' );
    ELSE
        DBMS_OUTPUT.PUT_LINE( 'Venduto <= 50k ' );
    END IF;
END;
/

```

```

CREATE OR REPLACE FUNCTION D06_GET_VENDUTO
(
    p_venduto IN number default 0
)

```

```

RETURN VARCHAR2
IS
    l_return VARCHAR2(100);
BEGIN
    IF p_venduto > 100000 THEN
        l_return := 'Venduto > 100k ';
    ELSIF p_venduto > 50000 THEN
        l_return := 'Venduto > 50k ';
    ELSE
        l_return := 'Venduto <= 50k ';
    END IF;
    RETURN l_return;
END;
/

```

```

DECLARE
    l_venduto VARCHAR2(100) := null;
BEGIN
    l_venduto := D06_GET_VENDUTO( p_venduto => 100000 );
    DBMS_OUTPUT.PUT_LINE( l_venduto );
END;
/

```

PROCEDURE E FUNZIONI

```
CREATE OR REPLACE PROCEDURE D06_VENDUTO ...  
...
```

```
CREATE OR REPLACE FUNCTION D06_GET_VENDUTO ...  
...
```

PACKAGE

```
CREATE OR REPLACE PACKAGE D06_PKG  
IS  
  PROCEDURE VENDUTO (  
    p_venduto IN number default 0  
  );  
  
  FUNCTION GET_VENDUTO (  
    p_venduto IN number default 0  
  )  
  
  RETURN VARCHAR2;  
  
END D06_PKG;  
/
```

Specifiche del Package

```

CREATE OR REPLACE PACKAGE D06_PKG
IS
  PROCEDURE VENDUTO (
    p_venduto IN number default 0
  );

  FUNCTION GET_VENDUTO (
    p_venduto IN number default 0
  )
    RETURN VARCHAR2;

END D06_PKG;
/

```

Corpo del Package

```

CREATE OR REPLACE PACKAGE BODY D06_PKG
IS
  PROCEDURE VENDUTO (
    p_venduto IN number default 0
  )
  IS
  BEGIN
    IF p_venduto > 100000 THEN
      DBMS_OUTPUT.PUT_LINE( 'Venduto > 100k ' );
    ELSIF p_venduto > 50000 THEN
      DBMS_OUTPUT.PUT_LINE( 'Venduto > 50k ' );
    ELSE
      DBMS_OUTPUT.PUT_LINE( 'Venduto <= 50k ' );
    END IF;
  END VENDUTO;

  FUNCTION GET_VENDUTO (
    p_venduto IN number default 0
  )
    RETURN VARCHAR2
  IS
    l_return VARCHAR2(100);
  BEGIN
    IF p_venduto > 100000 THEN
      l_return := 'Venduto > 100k ';
    ELSIF p_venduto > 50000 THEN
      l_return := 'Venduto > 50k ';
    ELSE
      l_return := 'Venduto <= 50k ';
    END IF;
    RETURN l_return;
  END GET_VENDUTO;

END D06_PKG;
/

```

Richiamo Procedura nel Package


```
BEGIN
  D06_PKG.VENDUTO(p_venduto => 100000);
END;
/
```

Richiamo Funzione nel Package

```
DECLARE
  l_venduto VARCHAR2(100) := null;
BEGIN
  l_venduto := D06_PKG.GET_VENDUTO(p_venduto => 100000);
  DBMS_OUTPUT.PUT_LINE( l_venduto );
END;
/
```

Do6 Dipendente Attività

D06 Dipendente Attività

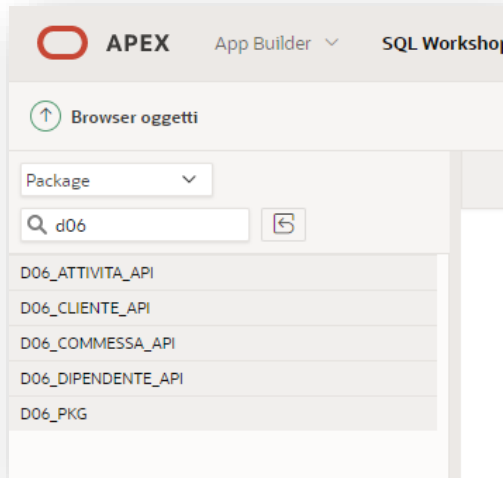
 **APEX** App Builder ▾ SQL Workshop ▾ Team Development ▾ Gallery

↑ Utilities \ Quick SQL

Quick SQL Clear Load Help Save Model **Generate SQL**

```
1 dipendente /insert 5
2   dipendente /nn /values francesca, marco, anna, simone, paola /unique
3   tariffa num /nn /values 200,150
4
5 cliente /insert 5
6   cliente vc50 /nn /values cliente_01,cliente_02,cliente_03,cliente_04,cliente_05 /unique
7
8 commessa /insert 5
9   commessa vc30 /nn /values COMM01, COMM02, COMM03, COMM04, COMM05 /unique
10
11 attivita /insert 200
12   dipendente_id /nn
13   cliente_id /nn
14   commessa_id vc100 /nn
15   data d /nn
16   ore num /nn
17
18 # settings = { prefix: "D06", ondelete: "RESTRICT", api: true, semantics: "CHAR", language: "EN", APEX: true }
```

D06 Dipendente Attività



```
create or replace package d06_cliente_api
is

    /* example:
    declare
        l_cliente          varchar2(10);
    begin
        d06_cliente_api.get_row (
            p_id              => 1,
            p_cliente         => l_cliente
        );
    end;
    */
    procedure get_row (
        p_id              in number,
        p_cliente         out varchar2
    );

    /* example:
    begin
        d06_cliente_api.insert_row (
            p_id              => null,
            p_cliente         => null
        );
    end;
    */
    procedure insert_row (
        p_id              in number default null,
        p_cliente         in varchar2 default null
    );
    procedure update_row (
        p_id              in number default null,
        p_cliente         in varchar2 default null
    );
    procedure delete_row (
        p_id              in number
    );
end d06_cliente_api;
```

```
create or replace package body d06_cliente_api
is

    procedure get_row (
        p_id              in number,
        p_cliente         out varchar2
    )
    is
    begin
        for c1 in (select * from d06_cliente where id = p_id) loop
            p_cliente := c1.cliente;
        end loop;
    end get_row;

    procedure insert_row (
        p_id              in number default null,
        p_cliente         in varchar2 default null
    )
    is
    begin
        insert into d06_cliente (
            id,
            cliente
        ) values (
            p_id,
            p_cliente
        );
    end insert_row;
    procedure update_row (
        p_id              in number default null,
        p_cliente         in varchar2 default null
    )
    is
    begin
        update d06_cliente set
            id = p_id,
            cliente = p_cliente
        where id = p_id;
    end update_row;
    procedure delete_row (
        p_id              in number
    )
    is
    begin
        delete from d06_cliente where id = p_id;
    end delete_row;
end d06_cliente_api;
```

D06 Dipendente Attività

Crea un'applicazione



Nome
D06 Dipendente Attività

Aspetto
Vita, Menu laterale



Pagine



Aggiungi pagina



Home

Vuota

Modifica



Dipendente

Report interattivo con form (d06_dipendente)

Modifica



Cliente

Report interattivo con form (d06_cliente)



Commissa

Report interattivo con form (d06_commissa)



Attività

Report interattivo con form (d06_attività)

Funzioni



Installa applicazione Web
progressiva



Pagina delle informazioni
Aggiunge una pagina di informazioni su
questa applicazione

Annulla

Crea applicazione

D06 Dipendente Attività

- Home
- Dipendente
- Cliente
- Commissa
- Attività



D06 Dipendente Attività

Dipendente

Cliente

Commissa

Attività

D06 Dipendente Attività

The screenshot displays the Oracle APEX App Builder interface. The main workspace shows a process named 'Elabora-form-Cliente' with steps: 'Dopo la sottomissione', 'Convalida in corso', 'Elaborazione', 'Dopo l'elaborazione', and 'Callback Ajax'. The 'Elaborazione' step is expanded, showing a 'Processi' section with 'Elabora-form-Cliente', 'Inserisci da Codice', and 'Chiudi finestra di dialogo'. The 'Inserisci da Codice' process is selected, showing its configuration: 'Nome' is 'Inserisci da Codice', 'Tipo' is 'Esegui codice', 'Area modificabile' is '- Seleziona -', 'Origine' is 'Database locale', 'Linguaggio' is 'PL/SQL', and the 'Codice PL/SQL' is a PL/SQL block. The 'Condizione lato server' is also visible, with 'Se si fa clic sul pulsante' set to 'CREATE' and 'Tipo' set to '- Seleziona -'.

/* example:

```
begin
  d06_cliente_api.insert_row (
    p_id
    p_cliente
  );
end;
```

*/

=> nul
=> nul

begin

```
d06_cliente_api.insert_row (
  p_cliente => :P5_CLIENTE
);
```

end;

Condizione lato server

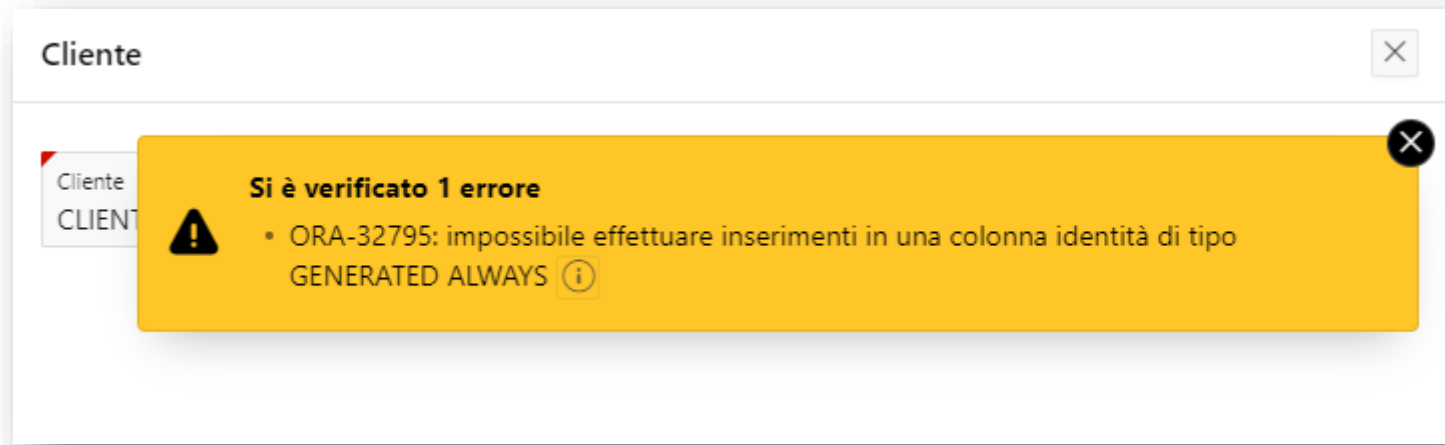
Se si fa clic sul
pulsante

CREATE

Tipo

- Seleziona -

D06 Dipendente Attività



```
alter table d06_cliente modify id generated always          as identity;  
alter table d06_cliente modify id generated by default on null as identity;
```

D06 Dipendente Attività

Identificazione

Nome	Modifica da Codice
Tipo	Esegui codice
Area modificabile	- Seleziona -

Origine

Posizione	Database locale
Linguaggio	PL/SQL

Codice PL/SQL

```
begin
    d06_cliente_api.update_row (
        p_id => :P5_ID,
        p_cliente => :P5_CLIENTE
    );
end;
```

Condizione lato server

Se si fa clic sul pulsante	SAVE
Tipo	- Seleziona -

```
/* example:
begin
    d06_cliente_api.insert_row (
        p_id
        p_cliente
    );
end;
*/
```

```
begin
    d06_cliente_api.update_row (
        p_id      => :P5_ID
        p_cliente => :P5_CLIENTE
    );
end;
```

D06 Dipendente Attività

The screenshot displays the Oracle APEX App Builder interface. The top navigation bar includes 'APEX', 'App Builder', 'SQL Workshop', 'Team Development', and 'Galleria'. The main workspace is divided into three panes: 'Page Designer', 'Process Designer', and a right-hand configuration panel.

Page Designer: Shows a page layout with a 'SUB REGIONS' section containing a 'NEXT' button. Below it is a 'DIALOG FOOTER' section with a 'Pulsanti' (Buttons) section containing 'PREVIOUS', 'CLOSE', 'DELETE', 'CANCEL', and 'DELETE' buttons. The 'REGION BODY' and 'CONTENUTO AREA' sections are also visible.

Process Designer: Shows a process flow with steps: 'Elabora form Cliente', 'Inserisci da Codice', 'Modifica da Codice' (highlighted), and 'Chiudi finestra di dialogo'. The 'Modifica da Codice' step is highlighted.

Right Panel: Contains configuration options for the selected step.

- Identificazione:** Nome: Modifica da Codice; Tipo: Esegui codice; Area modificabile: - Seleziona -.
- Origine:** Posizione: Database locale; Linguaggio: PL/SQL; Codice PL/SQL:

```
begin
  d06_cliente_api.update_row (
    p_id => :P5_ID,
    p_cliente => :P5_CLIENTE
  );
end;
```
- Condizione lato server:** Se si fa clic sul pulsante: SAVE; Tipo: - Seleziona -.

```
/* example:
begin
  d06_cliente_api.insert_row (
    p_id
    p_cliente
  );
end;
*/
```

```
begin
  d06_cliente_api.update_row (
    p_id      => :P5_ID
    p_cliente => :P5_CLIENTE
  );
end;
```

D06 Dipendente Attività

APEX App Builder | SQL Workshop | Team Development | Galleria | Cerca

Applicazione 762 \ Page Designer

Processo

- Dopo la sottomissione
- Convalida in corso
- Elaborazione
 - Processi
 - Elabora form Cliente
 - Inserisci da Codice
 - Modifica da Codice
 - Elimina da Codice**
 - Chiudi finestra di dialogo
- Dopo l'elaborazione
- Callback Ajax

Layout | Ricerca pagina | Guida

Identificazione

- Nome: Elimina da Codice
- Tipo: Esegui codice
- Area modificabile: - Seleziona -

Origine

- Posizione: Database locale
- Linguaggio: PL/SQL

Codice PL/SQL

```
begin
    d06_cliente_api.delete_row (
        p_id => :P5_ID
    );
end;
```

```
/* example:
begin
    d06_cliente_api.insert_row (
        p_id
        p_cliente
    );
end;
*/
```

```
=> null
=> null
begin
    d06_cliente_api.delete_row (
        p_id => :P5_ID
    );
end;
```

Condizione lato server

Se si fa clic sul pulsante

DELETE

Tipo

- Seleziona -

D06 Dipendente Attività

Applicazione 762 \ Page Designer

Layout Ricerca pagina Guida

Processo

Dopo la sottomissione

Convalida in corso

Elaborazione

Processi

Elabora form Cliente

Inserisci da Codice

Modifica da Codice

Elimina da Codice

Inserisci da Procedura

Chiudi finestra di dialogo

Dopo l'elaborazione

Callback Ajax

Cliente

DIALOG HEADER

CONTENT BODY

PREVIOUS

REGION BODY

AT P5_CLIENTE

CONTENUTO AREA

SUB REGIONS

NEXT

Aree

Elementi

Pulsanti

Identificazione

Nome

Inserisci da Procedura

Tipo

Richiama API

Area modificabile

- Seleziona -

Impostazioni

Tipo

PL/SQL Package

Proprietario

Parsing Schema

Package

D06_CLIENTE_API

Procedura o funzione

INSERT_ROW

Esecuzione

Sequenza

50

Punto

Elaborazione

Esegui processo

Una volta per visita nella pagina limposte

Applicazione 762 \ Page Designer

Layout Ricerca pagina Guida

Parametro

Dopo la sottomissione

Convalida in corso

Elaborazione

Processi

Elabora form Cliente

Inserisci da Codice

Modifica da Codice

Elimina da Codice

Inserisci da Procedura

Parametri

p_id

p_cliente

Chiudi finestra di dialogo

Dopo l'elaborazione

Cliente

DIALOG HEADER

CONTENT BODY

PREVIOUS

REGION BODY

AT P5_CLIENTE

CONTENUTO AREA

SUB REGIONS

NEXT

Identificazione

Nome

p_cliente

Parametro

Direzione

In

Tipo dati

VARCHAR2

Con impostazione predefinita

Valore

Tipo

Elemento

Elemento

P5_CLIENTE

Avanzate

Condizione lato server

Se si fa clic sul pulsante

CREATE

Tipo

- Seleziona -

D06 Dipendente Attività

Applicazione 762 \ Page Designer

Layout Ricerca pagina Guida

Dopo la sottomissione
Convalida in corso

Elaborazione

Processi

- Elabora form Cliente
- Inserisci da Codice
- Modifica da Codice
- Elimina da Codice
- Inserisci da Procedura
- Modifica da Procedura**
- Parametri
 - p_id
 - p_cliente
- Chiudi finestra di dialogo

Dopo l'elaborazione
Callback Ajax

Arece Elementi Pulsanti

Cliente

DIALOG HEADER

CONTENT BODY

PREVIOUS

REGION BODY

AI P5_CLIENTE

CONTENUTO AREA

SUB REGIONS

NEXT

Processo

Filtro

Identificazione

Nome Modifica da Procedura

Tipo Richiama API

Area modificabile - Seleziona -

Impostazioni

Tipo PL/SQL Package

Proprietario Parsing Schema

Package D06_CLIENTE_API

Procedura o funzione UPDATE_ROW

Esecuzione

Sequenza 60

Condizione lato server

Se si fa clic sul pulsante SAVE

Tipo - Seleziona -

Applicazione 762 \ Page Designer

Layout Ricerca pagina Guida

Dopo la sottomissione
Convalida in corso

Elaborazione

Processi

- Elabora form Cliente
- Inserisci da Codice
- Modifica da Codice
- Elimina da Codice
- Inserisci da Procedura
- Modifica da Procedura
- Elimina da Procedura**
- Chiudi finestra di dialogo

Dopo l'elaborazione
Callback Ajax

Arece Elementi Pulsanti

Cliente

DIALOG HEADER

CONTENT BODY

PREVIOUS

REGION BODY

AI P5_CLIENTE

CONTENUTO AREA

SUB REGIONS

NEXT

Processo

Filtro

Identificazione

Nome Elimina da Procedura

Tipo Richiama API

Area modificabile - Seleziona -

Impostazioni

Tipo PL/SQL Package

Proprietario Parsing Schema

Package D06_CLIENTE_API

Procedura o funzione DELETE_ROW

Esecuzione

Sequenza 70

Punto Elaborazione

Condizione lato server

Se si fa clic sul pulsante DELETE

Tipo - Seleziona -