



UNIVERSITÀ DI PISA

Guida Interfacce

Dario Ayrton Corveddu
Federico Giannotti
Francesco Nicolò
Stefano Piccoli

Laboratorio di basi di dati 2021-2022
Prof: Giovanna Rosone

7 gennaio 2022

Indice

1 Procedure	2
1.1 ApriPagina	2
1.2 Header	2
1.3 BannerUtente	2
1.4 Login	3
1.5 set_cookie	3
1.6 CreazioneSessione	3
1.7 RimozioneSessione	3
1.8 Bottone	4
1.9 ApriDiv	4
1.10 ChiudiDiv	4
1.11 Collegamento	5
1.12 ApriForm	5
1.13 ChiudiForm	5
1.14 InputText	6
1.15 InputTextArea	6
1.16 InputSubmit	6
1.17 InputDate	7
1.18 InputTime	7
1.19 InputNumber	7
1.20 SelectOpen	8
1.21 SelectOption	8
1.22 EmptySelectOption	8
1.23 SelectClose	9
1.24 InputRadioButton	9
1.25 InputCheckbox	9
1.26 InputCheckboxOnClick	10
1.27 ApriDivCard	10
1.28 InputReset	10
1.29 Redirect	11
1.30 ApriTabella	11
1.31 ChiudiTabella	11
1.32 apriRigaTabella	11
1.33 chiudiRigaTabella	12
1.34 apriElementoTabella	12
1.35 chiudiElementoTabella	12
1.36 ElementoTabella	12
1.37 intestazioneTabella	12
1.38 RedirectEsito	13

1.39 EsitoOperazione	13
2 Funzioni	14
2.1 get_id_sessione	14

1 Procedure

1.1 ApriPagina

Questa procedura permette di aprire i tag html e body di una pagina HTML e inserisce automaticamente lo stile predefinito.

\param titolo: il titolo da assegnare alla pagina
\param idSessione: la sessione dell'utente connesso

```
ApriPagina(  
    titolo varchar2 default 'Senza titolo',  
    idSessione int default 0  
);
```

1.2 Header

Questa procedura crea una testata nella pagina che include il menù a tendina e il banner utente.

```
procedure Header;
```

1.3 BannerUtente

Questa procedura crea il banner utente dove è possibile effettuare le operazioni di login e logout.

\param idSessione: la sessione dell'utente connesso

```
procedure BannerUtente (  
    idSessione int default 0  
);
```

1.4 Login

Questa procedura crea un pop-up dove l'utente può immettere le proprie credenziali ed effettuare l'accesso.

procedure Login;

1.5 set_cookie

Questa procedura crea un cookie nello storage locale del browser dell'utente dopo aver effettuato il login.

```
\param idSessione: la sessione dell'utente connesso
\param url: url nel quale reindirizzare l'utente dopo la creazione del cookie
procedure set_cookie (
    idSessione IN UTENTILOGIN.idUtenteLogin%TYPE,
    url VARCHAR2 DEFAULT '
');
```

1.6 CreazioneSessione

Questa procedura crea una sessione utente all'interno del DB.

```
\param usernames: nome utente che vuole effettuare il login
\param passwords: password utente che vuole effettuare il login
\param url: url nel quale reindirizzare l'utente dopo la creazione della sessione
```

```
procedure CreazioneSessione (
    usernames VARCHAR2 DEFAULT 'Sconosciuto',
    passwords VARCHAR2 DEFAULT 'Sconosciuto',
    url VARCHAR2 DEFAULT 'Sconosciuto'
);
```

1.7 RimozioneSessione

Questa procedura fa effettuare il logout all'utente distruggendo il cookie nello storage locale del browser e aggiornando il DB.

```
\param idSessione: la sessione dell'utente connesso
```

```
procedure RimozioneSessione(
    idsessione NUMBER DEFAULT 0
);
```

1.8 Bottone

Questa procedura crea un tag <button> nella pagina.

\param colore: specifica il colore del bottone utilizzando la stringa 'w3-' seguita dal colore in inglese

\param text: testo all'interno del bottone

\param id: id del bottone

\param: funzione javascript da richiamare quando si clicca sul bottone

```
procedure Bottone (  
    colore varchar2,  
    text varchar2 default 'myButton',  
    id varchar2 default '',  
    fun VARCHAR2 DEFAULT ''  
);
```

1.9 ApriDiv

Questa procedura crea un tag <div> nella pagina.

\param attributi: permette di specificare tutti gli attributi del div. Es: "class='w3-title' style='color:blue' id='titoloPrincipale'"

```
procedure ApriDiv (  
    attributi varchar2 default ''  
);
```

1.10 ChiudiDiv

Questa procedura crea un tag </div> nella pagina.

```
procedure ChiudiDiv;
```

1.11 Collegamento

Questa procedura crea un tag <a> con attributo href a forma di bottone. Permette all'utente di navigare tra le pagine del sito.

\param testo: testo all'interno del bottone
\param indirizzo: url nel quale verrà portato l'utente
\param classe: valore dell'attributo class
\param: funzione javascript da richiamare quando si clicca sul bottone

```
procedure Collegamento(  
    testo varchar2,  
    indirizzo varchar2,  
    classe varchar2 default '',  
    fun VARCHAR2 default ''  
);
```

1.12 ApriForm

Questa procedura crea un tag <form> nella pagina.

\param azione: procedura da eseguire una volta che l'utente invierà il form
\param nome: valore dell'attributo name
\param classe: valore dell'attributo class
\param root: se != 0, la procedura da eseguire si trova nel modgui1

```
procedure ApriForm(  
    azione varchar2,  
    nome varchar2 default 'myForm',  
    classe varchar2 default '',  
    root varchar2 default 0  
);
```

1.13 ChiudiForm

Questa procedura crea un tag </form> nella pagina.

```
procedure ChiudiForm;
```

1.14 InputText

Questa procedura crea un tag <input type="text"> nella pagina.

\param nome: valore dell'attributo name

\param placeholder: valore dell'attributo placeholder

\param required: se != 0, è obbligatorio compilare questo campo per inviare il form

\param valore: valore dell'attributo value

\param lunghezza: lunghezza massima della stringa da poter inserire nel campo

```
procedure InputText (  
    nome varchar2,  
    placeholder varchar2 default "",  
    required int default 0,  
    valore varchar2 default "",  
    lunghezza int default 1000  
);
```

1.15 InputTextArea

Questa procedura crea un tag <textarea> nella pagina

\param nome: valore dell'attributo name

\param required: se != 0, è obbligatorio compilare questo campo per inviare il form

\param valore: valore dell'attributo value

```
procedure InputTextArea (  
    nome varchar2,  
    placeholder varchar2 default "",  
    required int default 0,  
    valore varchar2 default ""  
);
```

1.16 InputSubmit

Questa procedura crea un tag <button> nella pagina

\param testo: valore dell'attributo testo, il quale comparirà come testo del bottone

```
procedure InputSubmit (  
    testo varchar2 default 'Submit'  
);
```

1.17 InputDate

Questa procedura crea un tag <input type="date"> nella pagina

\param id: valore dell'attributo id
\param nome: valore dell'attributo name
\param required: se != 0, è obbligatorio compilare questo campo per inviare il form
\param defaultValue: valore dell'attributo defaultValue

```
procedure InputDate (  
    id varchar2,  
    nome varchar2,  
    required int default 0,  
    defaultValue varchar2 default  
    "");
```

1.18 InputTime

Questa procedura crea un tag <input type="time"> nella pagina

\param id: valore dell'attributo id
\param nome: valore dell'attributo name
\param required: se != 0, è obbligatorio compilare questo campo per inviare il form
\param defaultValue: valore dell'attributo defaultValue

```
procedure InputTime (  
    id varchar2,  
    nome varchar2,  
    required int default 0,  
    defaultValue varchar2 default  
    "");
```

1.19 InputNumber

Questa procedura crea un tag <input type="number"> nella pagina

\param id: valore dell'attributo id
\param nome: valore dell'attributo name
\param required: se != 0, è obbligatorio compilare questo campo per inviare il form
\param defaultValue: valore dell'attributo defaultValue

```
procedure InputNumber (  
    id varchar2,  
    nome varchar2,  
    required int default 0,  
    defaultValue int default 0  
    );
```

1.20 SelectOpen

Questa procedura crea un tag <select> nella pagina

\param nome: valore dell'attributo name

\param id: valore dell'attributo id

```
procedure SelectOpen(  
    nome varchar2 default 'mySelect',  
    id varchar2 default 'mySelect'  
);
```

1.21 SelectOption

Questa procedura crea un tag <option> nella pagina, da utilizzare con SelectOpen e SelectClose

\param valore: valore dell'attributo valore

\param testo: valore dell'attributo testo

\param selected: se =1, l'opzione sarà quella di Default

```
procedure SelectOption(  
    valore varchar2,  
    testo varchar2 default 'Opzione',  
    selected int default 0  
);
```

1.22 EmptySelectOption

Questa procedura crea un tag <option> nella pagina, questa versione non permette di mettere valori al testo e al valore di submit

\param selected: se =1, l'opzione sarà quella di Default

```
procedure EmptySelectOption(  
    selected int default 0  
);
```

1.23 SelectClose

Questa procedura crea un tag `</select>` nella pagina.

```
procedure SelectClose;
```

1.24 InputRadioButton

Questa procedura crea un tag `<input type="radio">` nella pagina

\param testo: valore dell'attributo testo, che viene visualizzato sulla pagina

\param nome: valore dell'attributo nome

\param valore: valore dell'attributo valore, il valore effettivo che viene inviato

\param checked: se !=0, il radio verrà messo selezionato come default

\param disabled: se !=0, il radio verrà disabilitato e non interagibile

\param required: se != 0, è obbligatorio compilare questo campo per inviare il form

```
procedure InputRadioButton (  
    testo varchar2,  
    nome varchar2,  
    valore varchar2,  
    checked int default 0,  
    disabled int default 0,  
    required int default 0  
);
```

1.25 InputCheckbox

Questa procedura crea un tag `<input type="checkbox">`

\param testo: valore dell'attributo testo, che viene visualizzato sulla pagina

\param nome: valore dell'attributo nome

\param val: valore dell'attributo valore, il valore effettivo che viene inviato

\param disabled: se !=0, il radio verrà disabilitato e non interagibile

\param required: se != 0, è obbligatorio compilare questo campo per inviare il form

```

procedure InputCheckbox (
    testo varchar2,
    nome varchar2,
    checked int default 0,
    disabled int default 0,
    val varchar2 default 'on'
);

```

1.26 InputCheckboxOnClick

Questa procedura crea un tag <input type="checkbox">, nel quale come submit restituisce una funzione passata come parametro

\param testo: valore dell'attributo testo, che viene visualizzato sulla pagina
 \param nome: valore dell'attributo nome
 \param fun: valore della funzione da passare
 \param id: valore dell'attributo id
 \param disabled: se !=0, il radio verrà disabilitato e non interagibile
 \param required: se != 0, è obbligatorio compilare questo campo per inviare il form

```

procedure InputCheckboxOnClick (
    testo varchar2,
    nome varchar2,
    fun varchar2,
    id varchar2,
    checked int default 0,
    disabled int default 0
);

```

1.27 ApriDivCard

Questa procedura apre un tag <div> di tipo "we-card". Al suo interno può essere inserito un form o un elemento del DB.

```

procedure ApriDivCard;

```

1.28 InputReset

Questa procedura resetta tutti i campi all'interno del form.

```
procedure InputReset;
```

1.29 Redirect

Questa procedura reindirizza l'utente verso una nuova pagina.

\param destinazione: url nel quale l'utente viene reindirizzato

```
procedure Redirect (  
    destinazione varchar2  
);
```

1.30 ApriTabella

Questa procedura crea un tag <table> nella pagina.

\param classe: valore dell'attributo class

```
procedure apriTabella(  
    classe varchar2 default 'defTable'  
);
```

1.31 ChiudiTabella

Questa procedura crea un tag </table> nella pagina.

```
procedure ChiudiTabella;
```

1.32 apriRigaTabella

Questa procedura crea un tag <tr> nella pagina. Da usare all'interno di una tabella.

\param classe: valore dell'attributo class

```
procedure apriRigaTabella(  
    classe varchar2 default 'defRowTable'  
);
```

1.33 chiudiRigaTabella

Questa procedura crea un tag `</tr>` nella pagina. Da usare all'interno di una tabella.
procedure chiudiRigaTabella;

1.34 apriElementoTabella

Questa procedura crea un tag `<td>` nella pagina. Da usare all'interno di una tabella.

\param classe: valore dell'attributo class

\param id: valore dell'attributo id

```
procedure apriElementoTabella(  
    classe varchar2 default 'defElementoTabella',  
    id varchar2 default ''  
);
```

1.35 chiudiElementoTabella

Questa procedura crea un tag `</td>` nella pagina. Da usare all'interno di una tabella.
procedure chiudiElementoTabella;

1.36 ElementoTabella

Questa procedura crea un testo da utilizzare dentro un campo della tabella.

\param testo: testo da inserire nell'elemento

```
procedure ElementoTabella(  
    testo varchar2  
);
```

1.37 intestazioneTabella

Questa procedura crea un tag `<th></th>` nel quale è possibile inserire un testo. Da usare all'interno di una tabella.

\param testo: testo da inserire nell'elemento

\param classe: valore dell'attributo class

```
procedure intestazioneTabella(  
    classe varchar2 default 'defIntestazioneTabella',  
    testo varchar2  
);
```

```

testo varchar2,
classe varchar2 default 'defHeaderTable'
);

```

1.38 RedirectEsito

Questa procedura reindirizza l'utente verso una nuova pagina. Si può personalizzare la nuova pagina passando i parametri in questa procedura.

```

\param pageTitle: titolo della pagina
\param msg: messaggio nel sottotesto
\param nuovaOp: testo del bottone che riporta ad una nuova procedura
\param nuovaOpURL: procedura da richiamare quando si clicca sul bottone
\param parametrinuovaOp: parametri della procedura da richiamare
\param backToMenu: testo del bottone che riporta al menù
\param backToMenuURL: procedura da richiamare quando si clicca sul bottone menù
\param paramtribackToMenu: parametri della procedura menù da richiamare

```

```

procedure RedirectEsito (
    pageTitle VARCHAR2 DEFAULT NULL,
    msg VARCHAR2 DEFAULT NULL,
    nuovaOp VARCHAR2 DEFAULT NULL,
    nuovaOpURL VARCHAR2 DEFAULT NULL,
    parametrinuovaOp VARCHAR2 DEFAULT "",
    backToMenu VARCHAR2 DEFAULT NULL,
    backToMenuURL VARCHAR2 DEFAULT NULL,
    paramtribackToMenu VARCHAR2 DEFAULT ""
);

```

1.39 EsitoOperazione

Questa procedura crea una pagina personalizzata per l'utente. Viene richiamata dalla procedura RedirectEsito.

```

\param pageTitle: titolo della pagina
\param msg: messaggio nel sottotesto
\param nuovaOp: testo del bottone che riporta ad una nuova procedura
\param nuovaOpURL: procedura da richiamare quando si clicca sul bottone
\param parametrinuovaOp: parametri della procedura da richiamare
\param backToMenu: testo del bottone che riporta al menù
\param backToMenuURL: procedura da richiamare quando si clicca sul bottone menù

```

\param paramtribackToMenu: parametri della procedura menù da richiamare

```
procedure RedirectEsito (  
  pageTitle VARCHAR2 DEFAULT NULL,  
  msg VARCHAR2 DEFAULT NULL,  
  nuovaOp VARCHAR2 DEFAULT NULL,  
  nuovaOpURL VARCHAR2 DEFAULT NULL,  
  parametrinuovaOp VARCHAR2 DEFAULT "",  
  backToMenu VARCHAR2 DEFAULT NULL,  
  backToMenuURL VARCHAR2 DEFAULT NULL,  
  paramtribackToMenu VARCHAR2 DEFAULT ""  
);
```

2 Funzioni

2.1 get_id_sessione

Questa funzione restituisce l'id della sessione corrente andando a controllare il relativo cookie nello storage locale del browser.

```
function get_id_sessione return NUMBER;
```