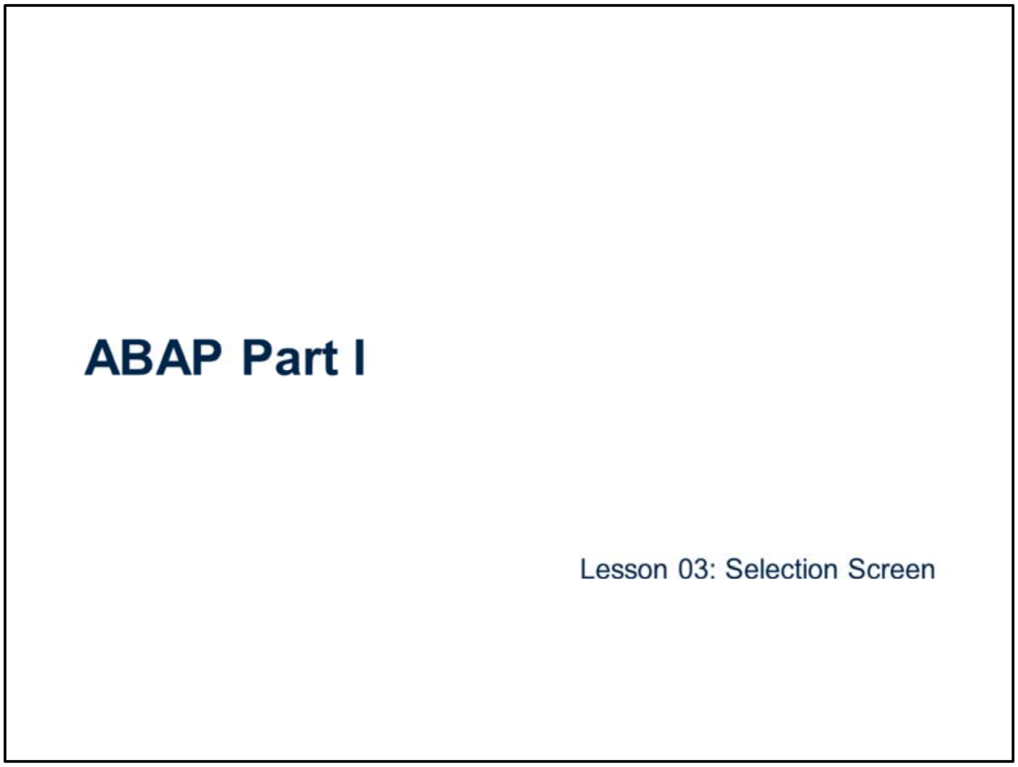
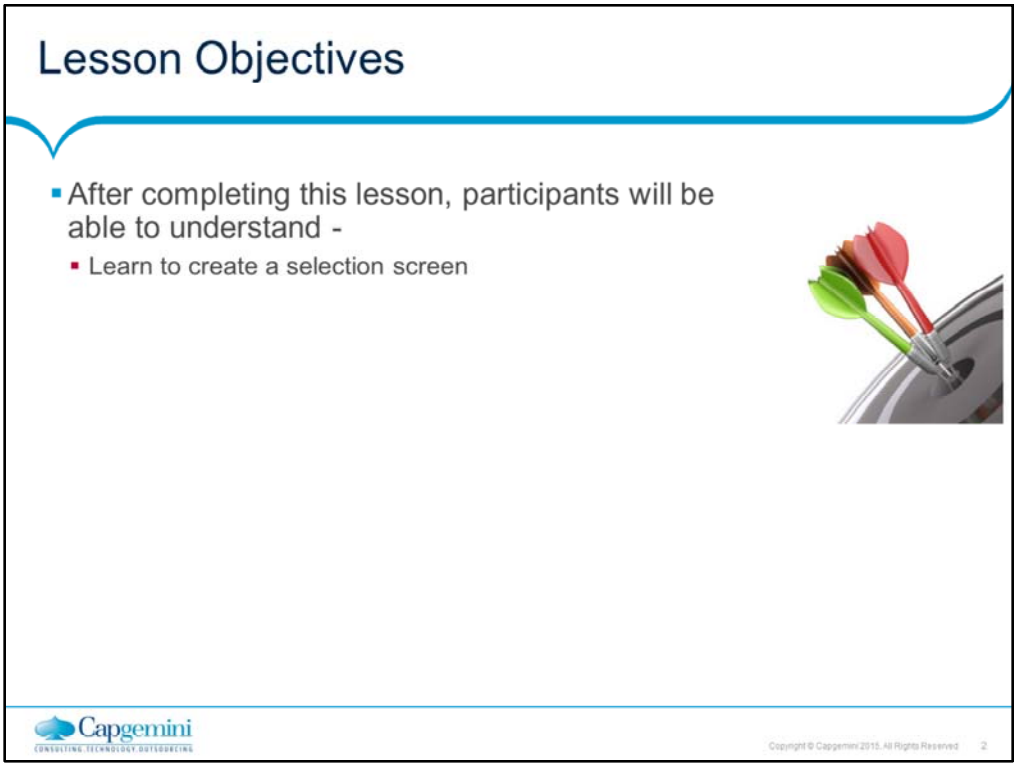


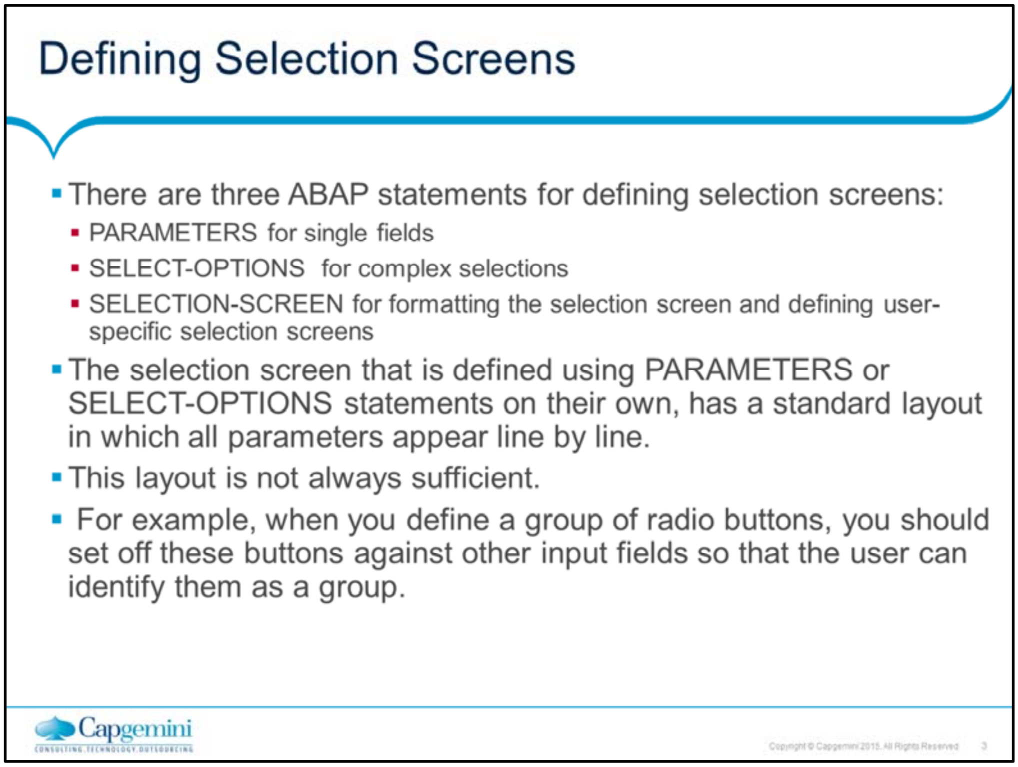
ABAP Part I

Lesson 03: Selection Screen



Lesson Objectives

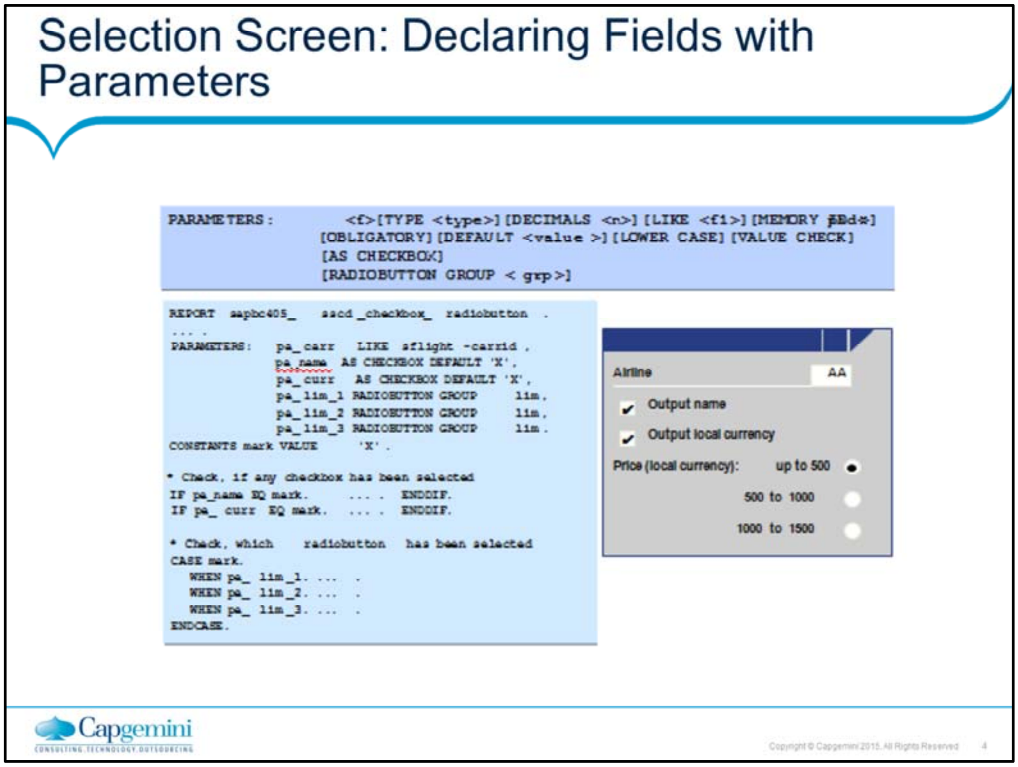
- After completing this lesson, participants will be able to understand -
  - Learn to create a selection screen



Defining Selection Screens

- There are three ABAP statements for defining selection screens:
  - PARAMETERS for single fields
  - SELECT-OPTIONS for complex selections
  - SELECTION-SCREEN for formatting the selection screen and defining user-specific selection screens
- The selection screen that is defined using PARAMETERS or SELECT-OPTIONS statements on their own, has a standard layout in which all parameters appear line by line.
- This layout is not always sufficient.
- For example, when you define a group of radio buttons, you should set off these buttons against other input fields so that the user can identify them as a group.

When defining input screens for the use of reporting on data, various selection elements can be combined to maintain data integrity and still be easy to use. These elements include standard input fields such as selection options and parameters, as well as others such as check boxes, radio buttons, and pushbuttons all arranged on a selection screen



Selection Screen: Declaring Fields with Parameters

```
PARAMETERS :        <f>[TYPE <type>][DECIMALS <n>][LIKE <f1>][MEMORY pBd*]
                    [OBLIGATORY][DEFAULT <value >][LOWER CASE][VALUE CHECK]
                    [AS CHECKBOX]
                    [RADIOBUTTON GROUP < grp>]
```

```
REPORT  sapbc405_   sscd _checkbox_  radiobutton  .
.... .
PARAMETERS:   pa_carr   LIKE  sflight -carrid ,
              pa_name  AS CHECKBOX DEFAULT 'X',
              pa_curr   AS CHECKBOX DEFAULT 'X',
              pa_lim_1 RADIOBUTTON GROUP     lim,
              pa_lim_2 RADIOBUTTON GROUP     lim,
              pa_lim_3 RADIOBUTTON GROUP     lim .
CONSTANTS mark VALUE       'X' .

* Check, if any checkbox has been selected
IF pa_name EQ mark.      .... .  ENDDIF.
IF pa_ curr  EQ mark.   .... .  ENDDIF.

* Check, which    radiobutton   has been selected
CASE mark.
   WHEN pa_ lim_1. ...   .
   WHEN pa_ lim_2. ...   .
   WHEN pa_ lim_3. ...   .
ENDCASE.
```

Airline AA

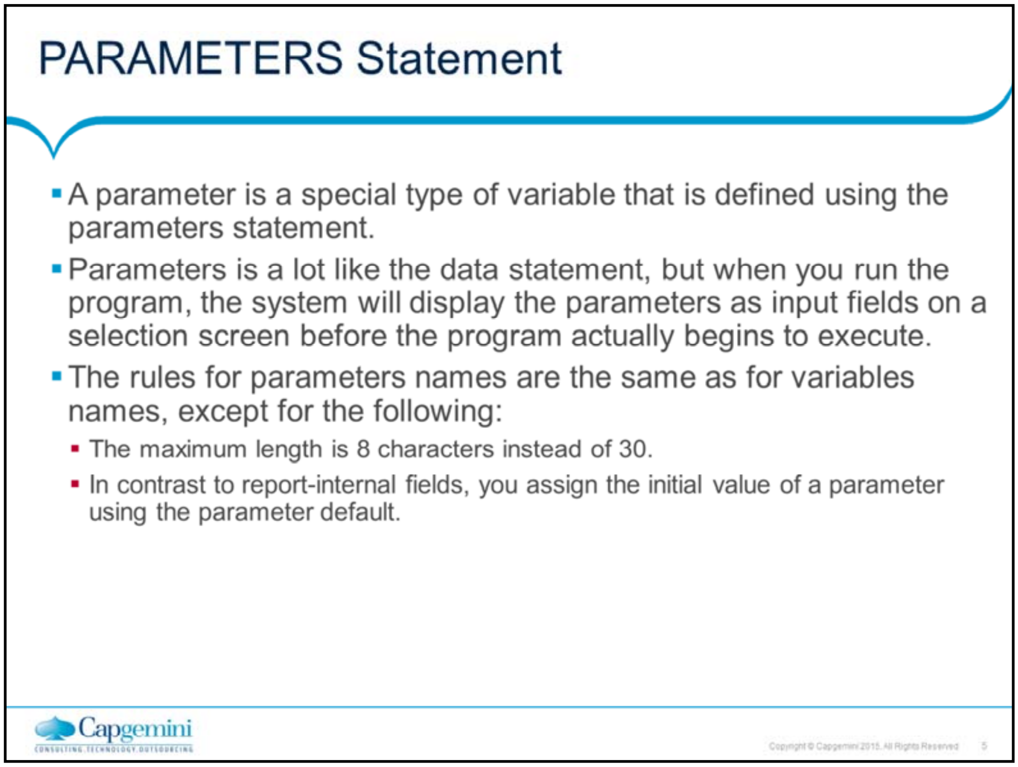
☑ Output name

☑ Output local currency

Price (local currency): up to 500 ●

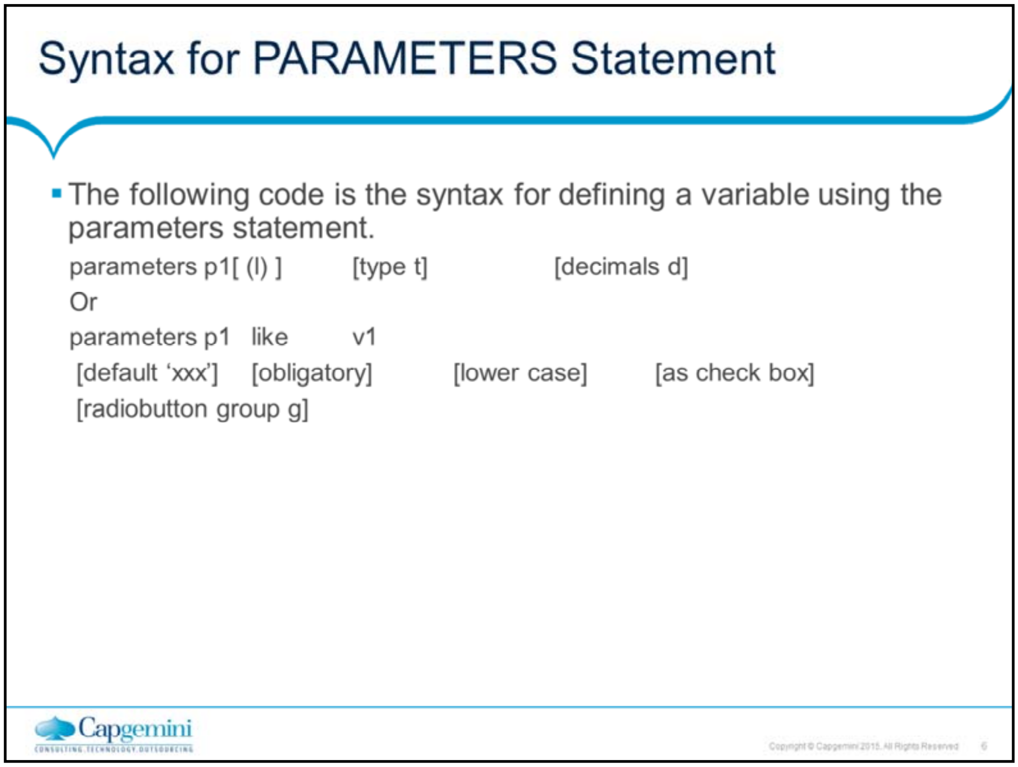
500 to 1000 ○

1000 to 1500 ○



PARAMETERS Statement

- A parameter is a special type of variable that is defined using the parameters statement.
- Parameters is a lot like the data statement, but when you run the program, the system will display the parameters as input fields on a selection screen before the program actually begins to execute.
- The rules for parameters names are the same as for variables names, except for the following:
  - The maximum length is 8 characters instead of 30.
  - In contrast to report-internal fields, you assign the initial value of a parameter using the parameter default.



Syntax for PARAMETERS Statement

- The following code is the syntax for defining a variable using the parameters statement.

```
parameters p1[ (l) ]        [type t]             [decimals d]
Or
parameters p1   like        v1
 [default ‘xxx’]   [obligatory]         [lower case]         [as check box]
 [radiobutton group g]
```

Where:

p1 is the parameter name

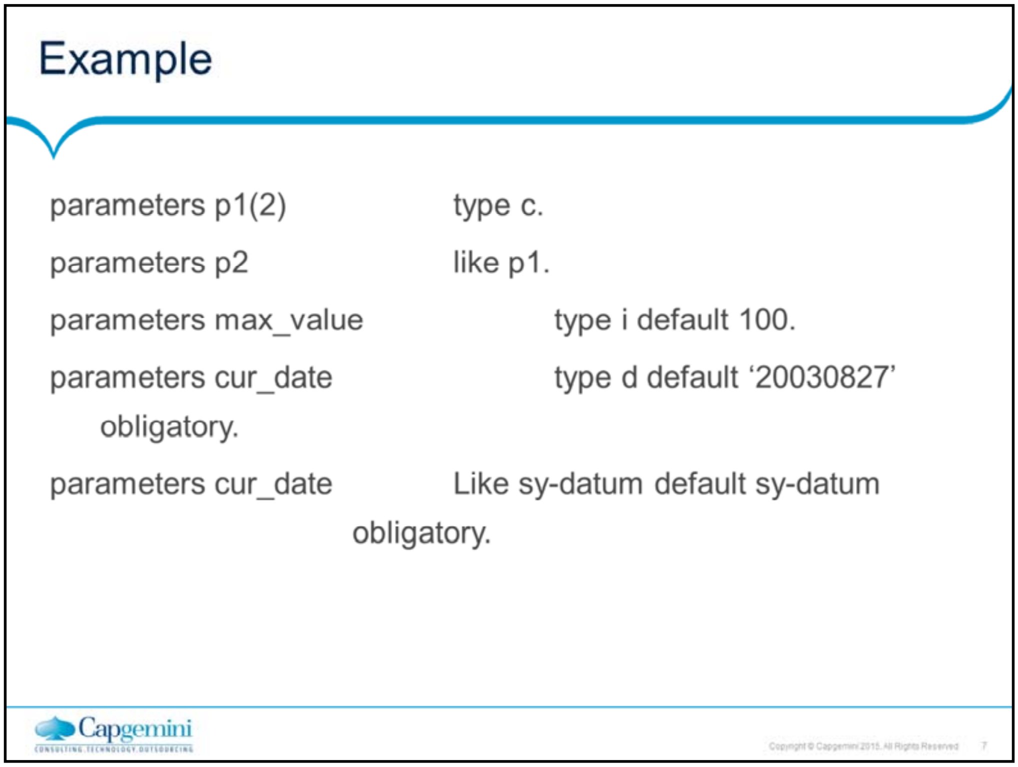
v1 is the name of a previously defined variable or parameter, or is the name of a field that belongs to a table or structure in the Data Dictionary

(l) is the internal length specification

t is the data type

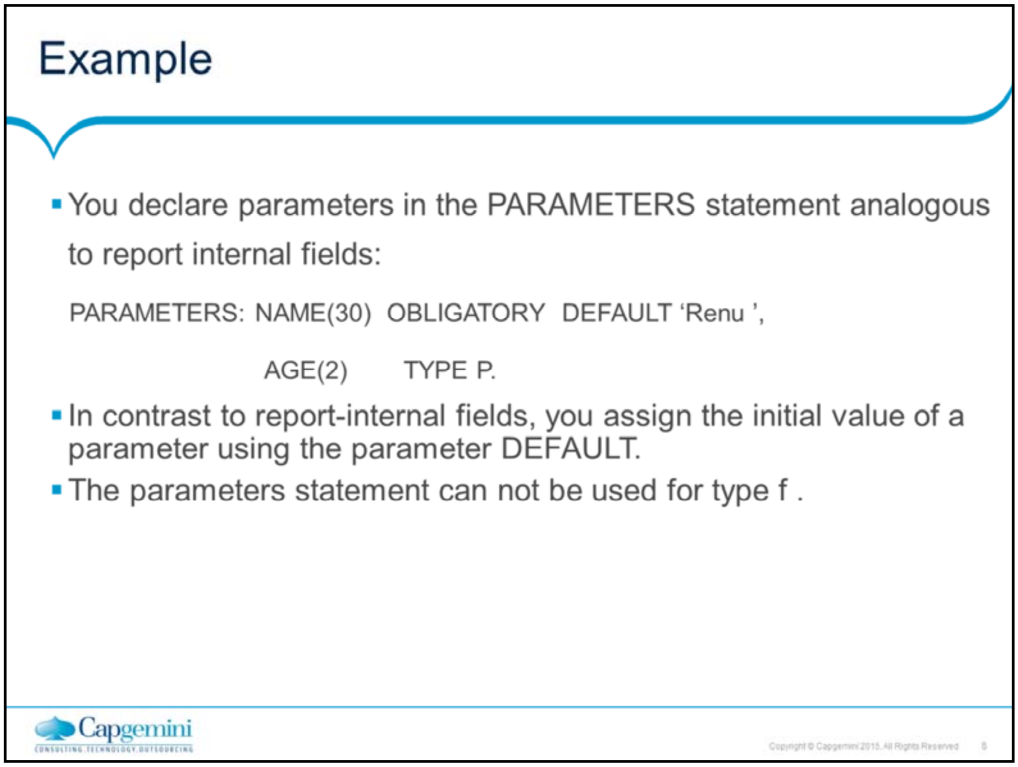
d is the number of decimal places (used only with type p)

‘xxx’ is a literal or previously defined variable that supplies a default value



Example

```
parameters p1(2)              type c.
parameters p2                 like p1.
parameters max_value                 type i default 100.
parameters cur_date                  type d default ‘20030827’
    obligatory.
parameters cur_date           Like sy-datum default sy-datum
                       obligatory.
```

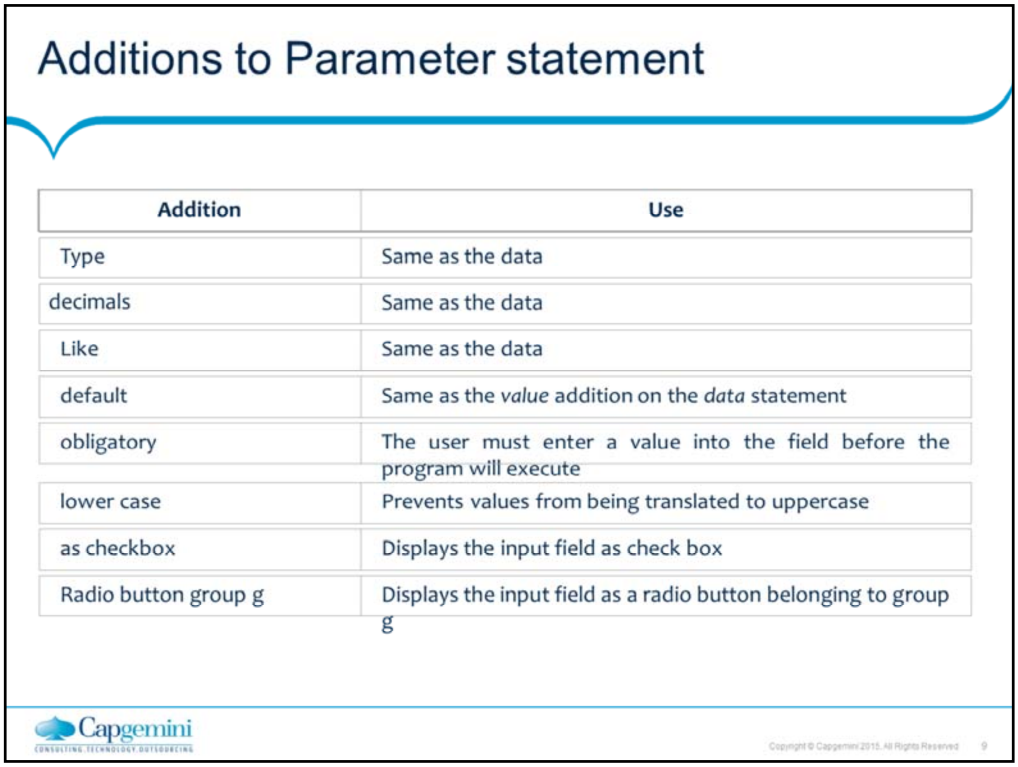


Example

- You declare parameters in the PARAMETERS statement analogous to report internal fields:

```
PARAMETERS: NAME(30)  OBLIGATORY  DEFAULT 'Renu ',
                      AGE(2)        TYPE P.
```

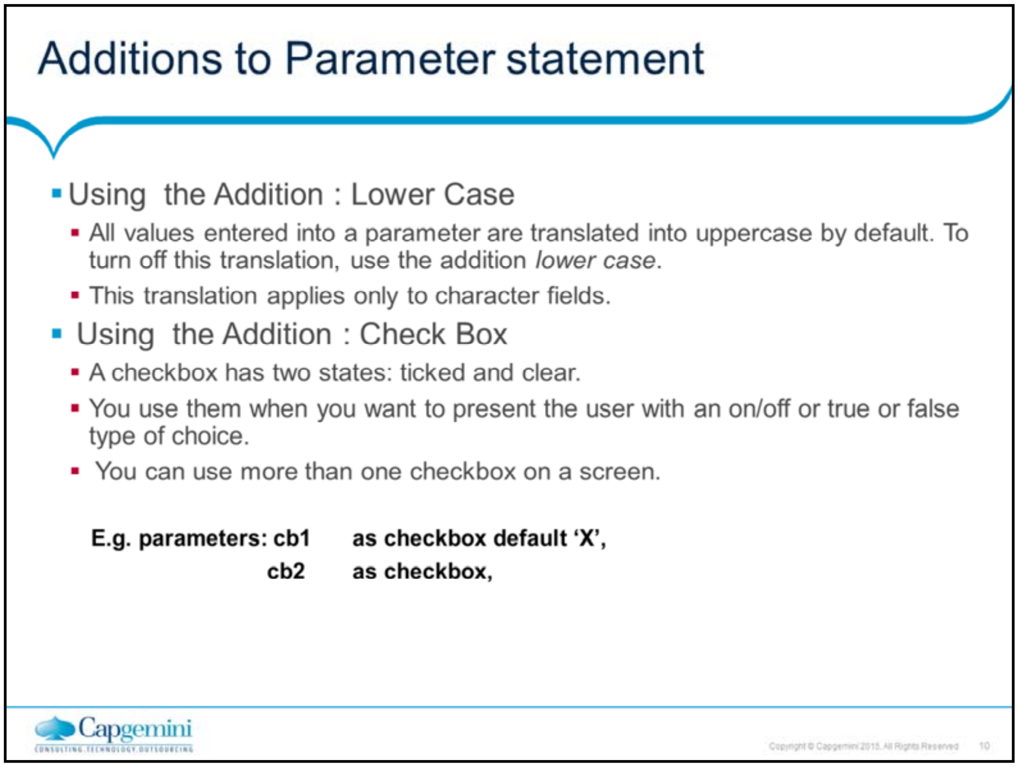
- In contrast to report-internal fields, you assign the initial value of a parameter using the parameter DEFAULT.
- The parameters statement can not be used for type f .



Additions to Parameter statement

Addition	Use
Type	Same as the data
decimals	Same as the data
Like	Same as the data
default	Same as the the *value* addition on the *data* statement
obligatory	The user must enter a value into the field before the program will execute
lower case	Prevents values from being translated to uppercase
as checkbox	Displays the input field as check box
Radio button group g	Displays the input field as a radio button belonging to group g

9

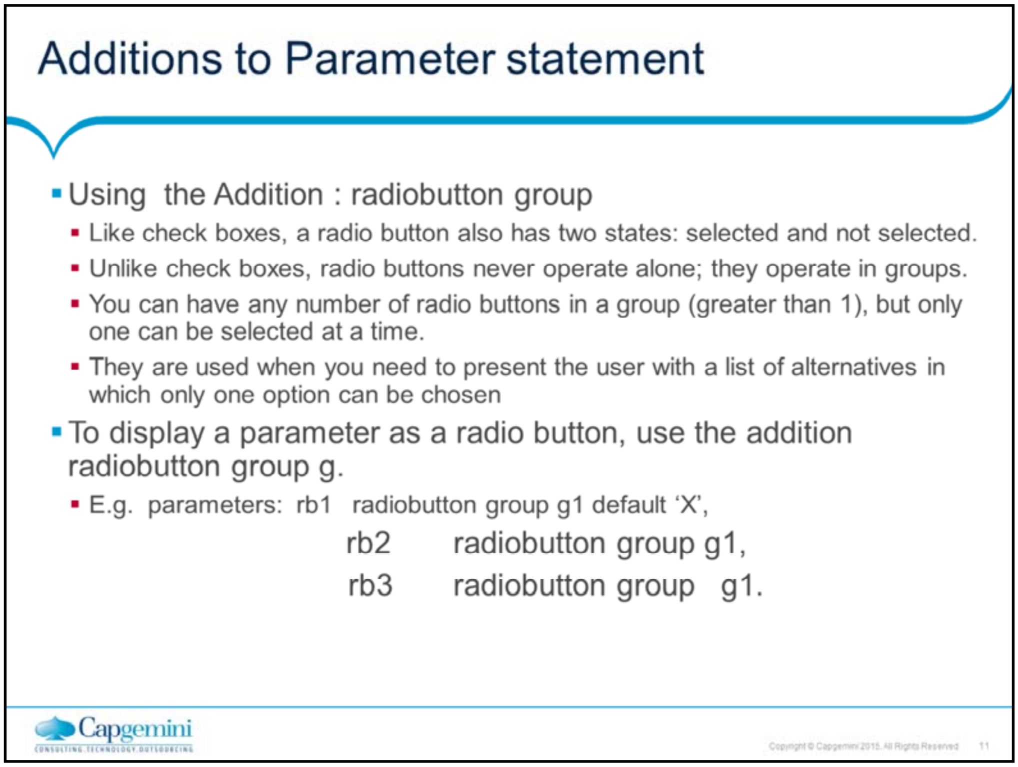


Additions to Parameter statement

- Using the Addition : Lower Case
  - All values entered into a parameter are translated into uppercase by default. To turn off this translation, use the addition *lower case*.
  - This translation applies only to character fields.
- Using the Addition : Check Box
  - A checkbox has two states: ticked and clear.
  - You use them when you want to present the user with an on/off or true or false type of choice.
  - You can use more than one checkbox on a screen.

```
E.g. parameters: cb1      as checkbox default 'X',
                 cb2      as checkbox,
```

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING



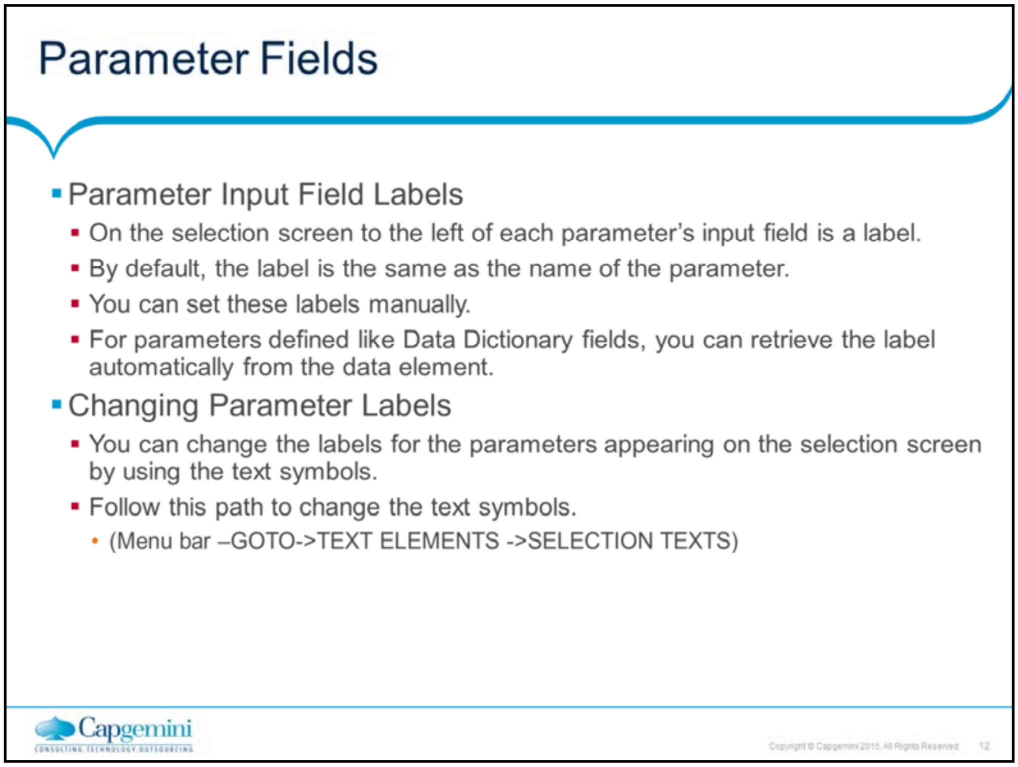
Additions to Parameter statement

- Using the Addition : radiobutton group
  - Like check boxes, a radio button also has two states: selected and not selected.
  - Unlike check boxes, radio buttons never operate alone; they operate in groups.
  - You can have any number of radio buttons in a group (greater than 1), but only one can be selected at a time.
  - They are used when you need to present the user with a list of alternatives in which only one option can be chosen
- To display a parameter as a radio button, use the addition radiobutton group g.
  - E.g.

```
parameters:  rb1   radiobutton group g1 default ‘X’,
             rb2       radiobutton group g1,
             rb3       radiobutton group   g1.
```

11

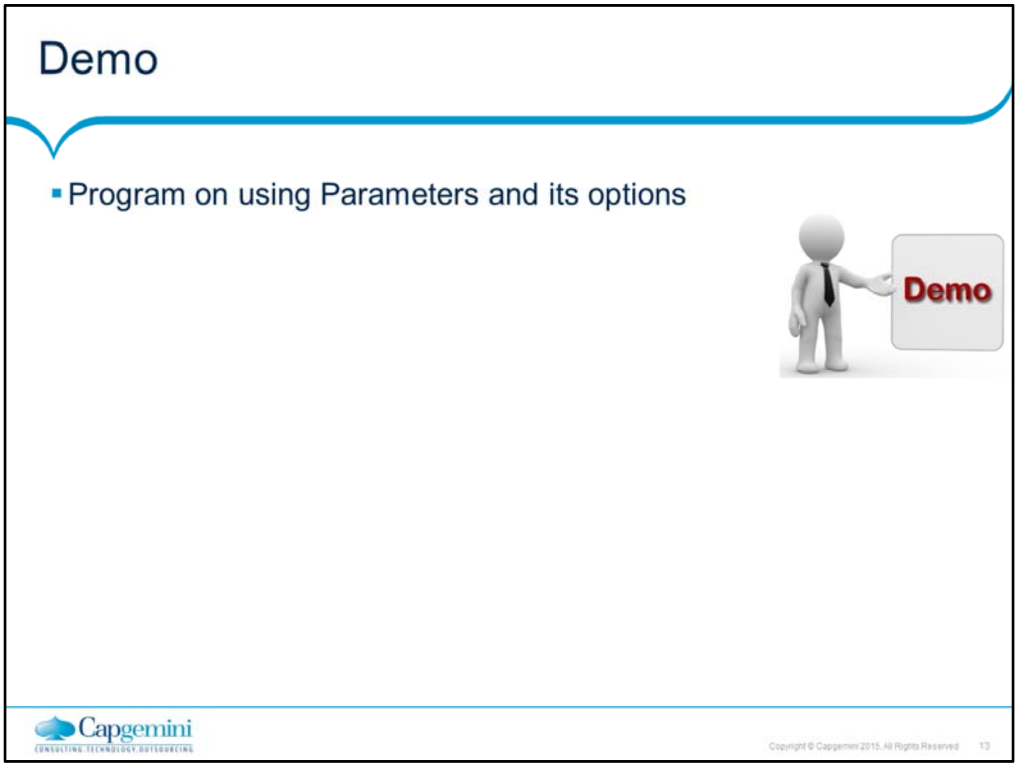
To display a parameter as a radio button, use the addition radiobutton group g. You cannot specify a data type or length; it will default to type c and length 1. g is an arbitrary group name one to four characters long. You can have more than one group in a program. The parameter will contain a capital X, if the radio button is selected; it will contain a blank if not selected. To be initially selected, the radio button should contain a default value of capital X. No other values are valid for a radio button. E



Parameter Fields

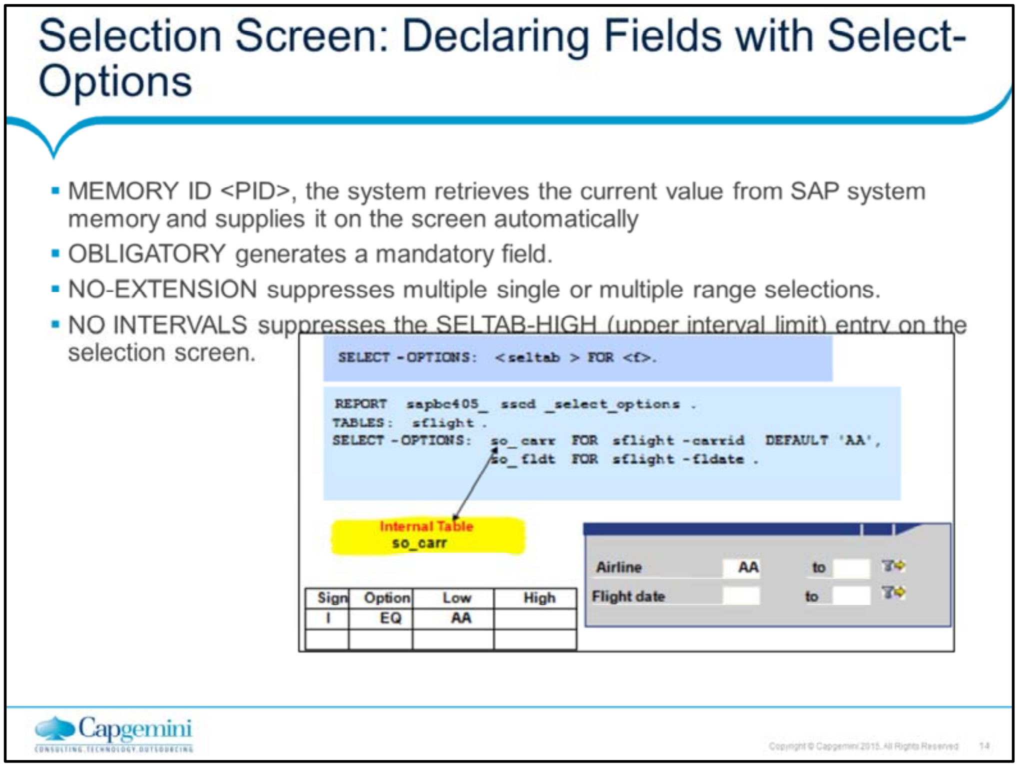
- Parameter Input Field Labels
  - On the selection screen to the left of each parameter’s input field is a label.
  - By default, the label is the same as the name of the parameter.
  - You can set these labels manually.
  - For parameters defined like Data Dictionary fields, you can retrieve the label automatically from the data element.
- Changing Parameter Labels
  - You can change the labels for the parameters appearing on the selection screen by using the text symbols.
  - Follow this path to change the text symbols.
    - (Menu bar –GOTO->TEXT ELEMENTS ->SELECTION TEXTS)

12



Demo

- Program on using Parameters and its options



Selection Screen: Declaring Fields with Select-Options

- MEMORY ID <PID>, the system retrieves the current value from SAP system memory and supplies it on the screen automatically
- OBLIGATORY generates a mandatory field.
- NO-EXTENSION suppresses multiple single or multiple range selections.
- NO INTERVALS suppresses the SELTAB-HIGH (upper interval limit) entry on the selection screen.

```
SELECT -OPTIONS:  <seltab > FOR <f>.
```

```
REPORT  sapbc405_ sscd _select_options .
TABLES:  sflight .
SELECT -OPTIONS:  so_ carr  FOR  sflight -carrid  DEFAULT 'AA',
                  so_ fldt  FOR  sflight -fldate .
```

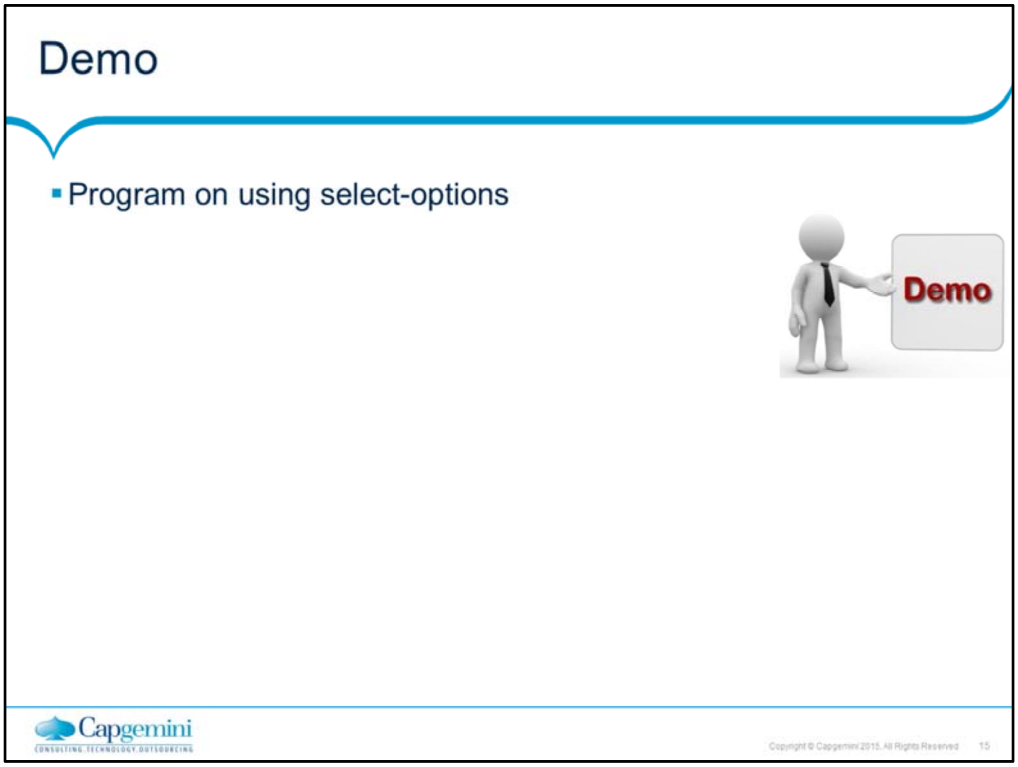
Internal Table
so_carr

Sign	Option	Low	High
I	EQ	AA	

Airline AA to

Flight date to

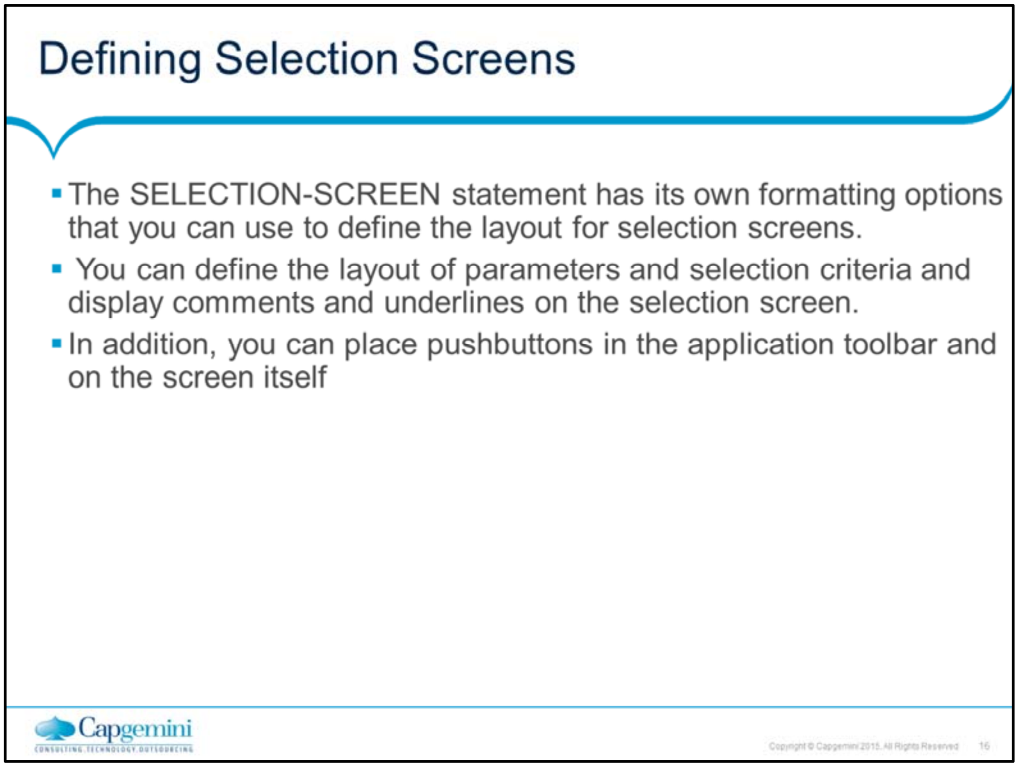
14



Demo

- Program on using select-options

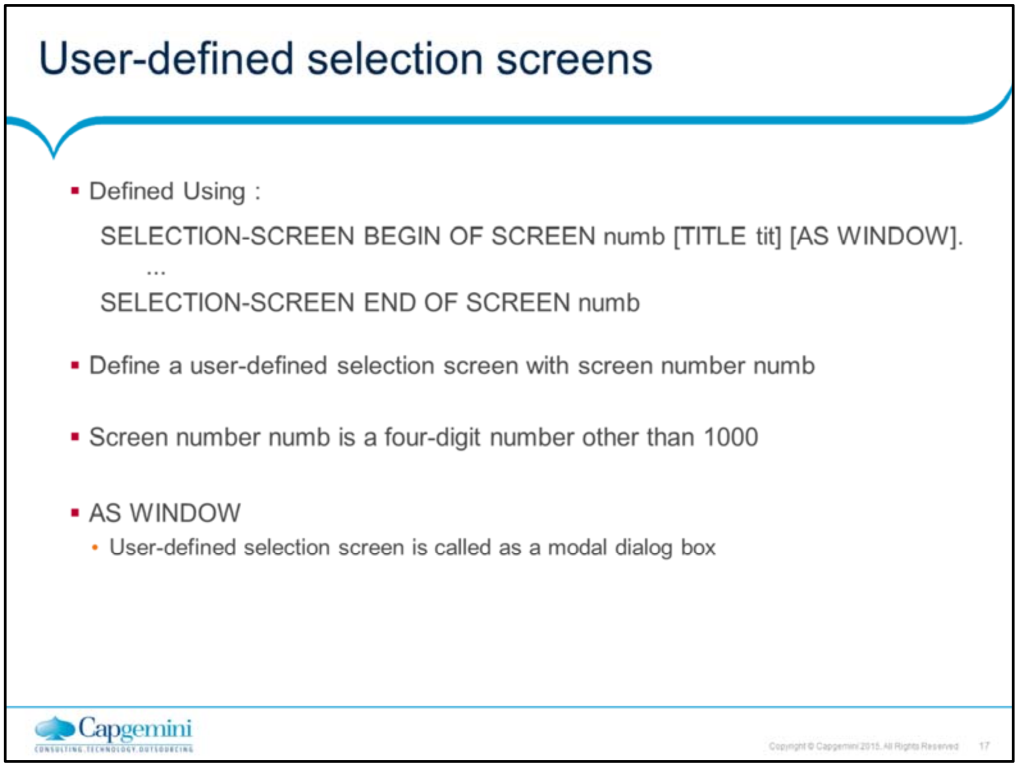
15



Defining Selection Screens

- The SELECTION-SCREEN statement has its own formatting options that you can use to define the layout for selection screens.
- You can define the layout of parameters and selection criteria and display comments and underlines on the selection screen.
- In addition, you can place pushbuttons in the application toolbar and on the screen itself

16



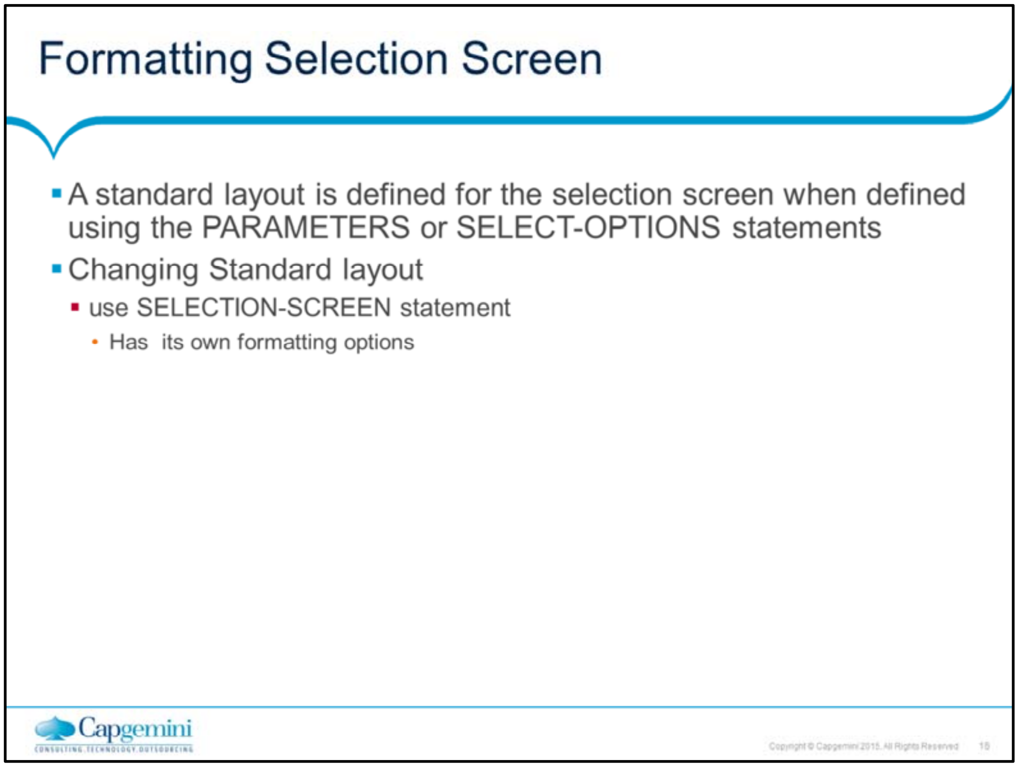
User-defined selection screens

- Defined Using :

```
SELECTION-SCREEN BEGIN OF SCREEN numb [TITLE tit] [AS WINDOW].
      ...
SELECTION-SCREEN END OF SCREEN numb
```

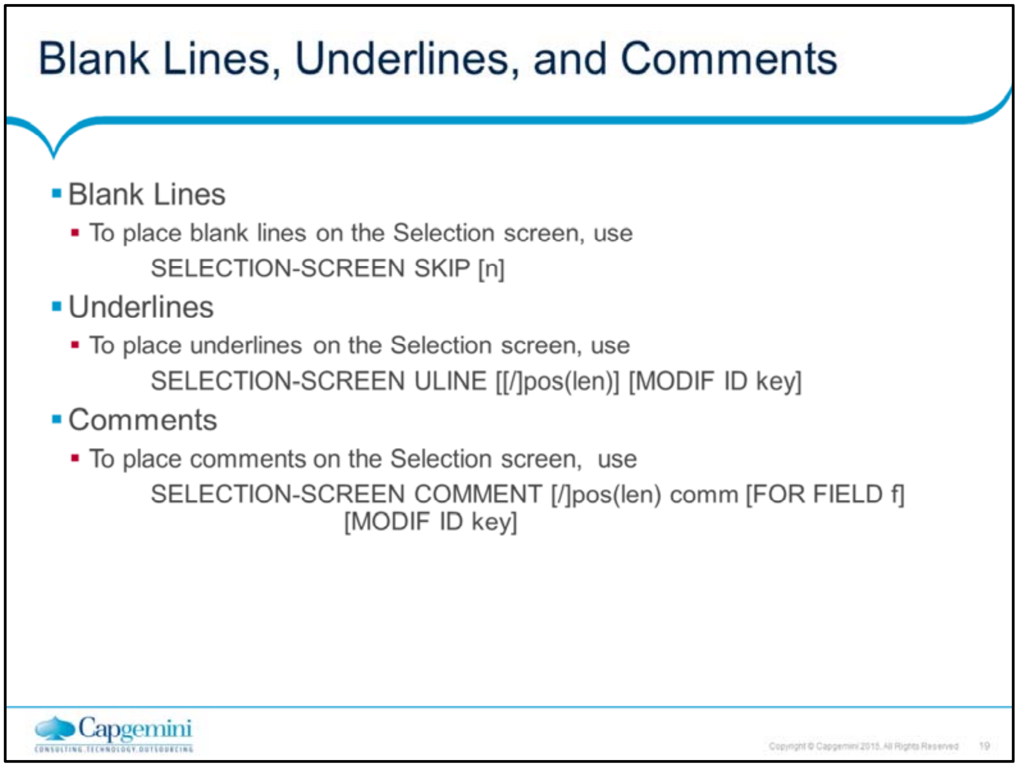
- Define a user-defined selection screen with screen number numb
- Screen number numb is a four-digit number other than 1000
- AS WINDOW
  - User-defined selection screen is called as a modal dialog box

17



Formatting Selection Screen

- A standard layout is defined for the selection screen when defined using the PARAMETERS or SELECT-OPTIONS statements
- Changing Standard layout
  - use SELECTION-SCREEN statement
    - Has its own formatting options



Blank Lines, Underlines, and Comments

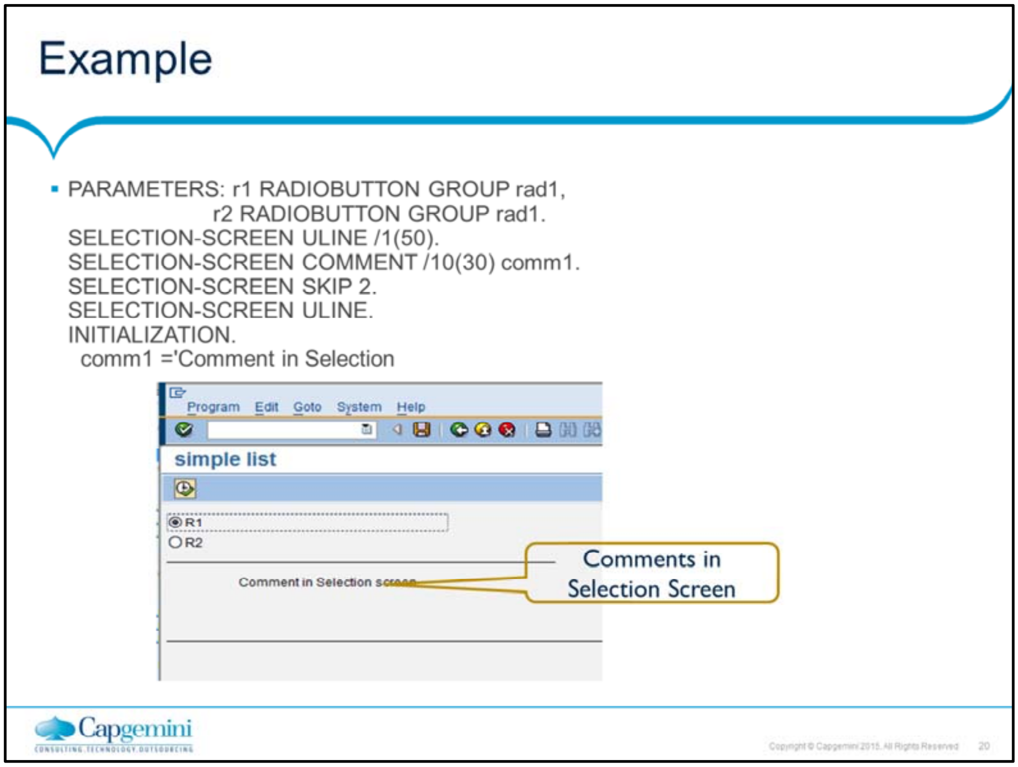
- Blank Lines
  - To place blank lines on the Selection screen, use

    SELECTION-SCREEN SKIP [n]
- Underlines
  - To place underlines on the Selection screen, use

    SELECTION-SCREEN ULINE [[/]pos(len)] [MODIF ID key]
- Comments
  - To place comments on the Selection screen, use

    SELECTION-SCREEN COMMENT [/]pos(len) comm [FOR FIELD f] [MODIF ID key]

19



Example

```
PARAMETERS: r1 RADIOBUTTON GROUP rad1,
            r2 RADIOBUTTON GROUP rad1.
SELECTION-SCREEN ULINE /1(50).
SELECTION-SCREEN COMMENT /10(30) comm1.
SELECTION-SCREEN SKIP 2.
SELECTION-SCREEN ULINE.
INITIALIZATION.
 comm1 ='Comment in Selection
```

Program Edit Goto System Help
simple list

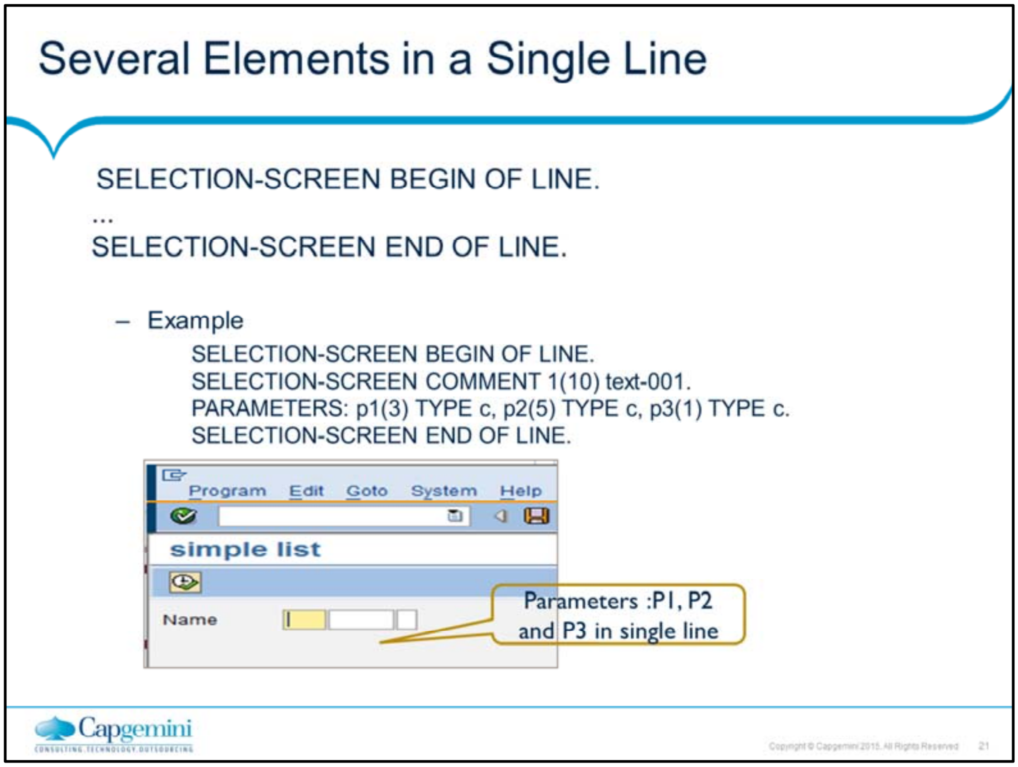
● R1
○ R2

Comment in Selection screen

Comments in Selection Screen

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

selection-screen skip n - This statement creates a blank line for as many lines for n lines on the selection screen
selection-screen uline -This will place an underline on the screen at a specified location for a specified length.
selection-screen comment -This will place a comment on the selection screen.



Several Elements in a Single Line

SELECTION-SCREEN BEGIN OF LINE.
...
SELECTION-SCREEN END OF LINE.

- Example

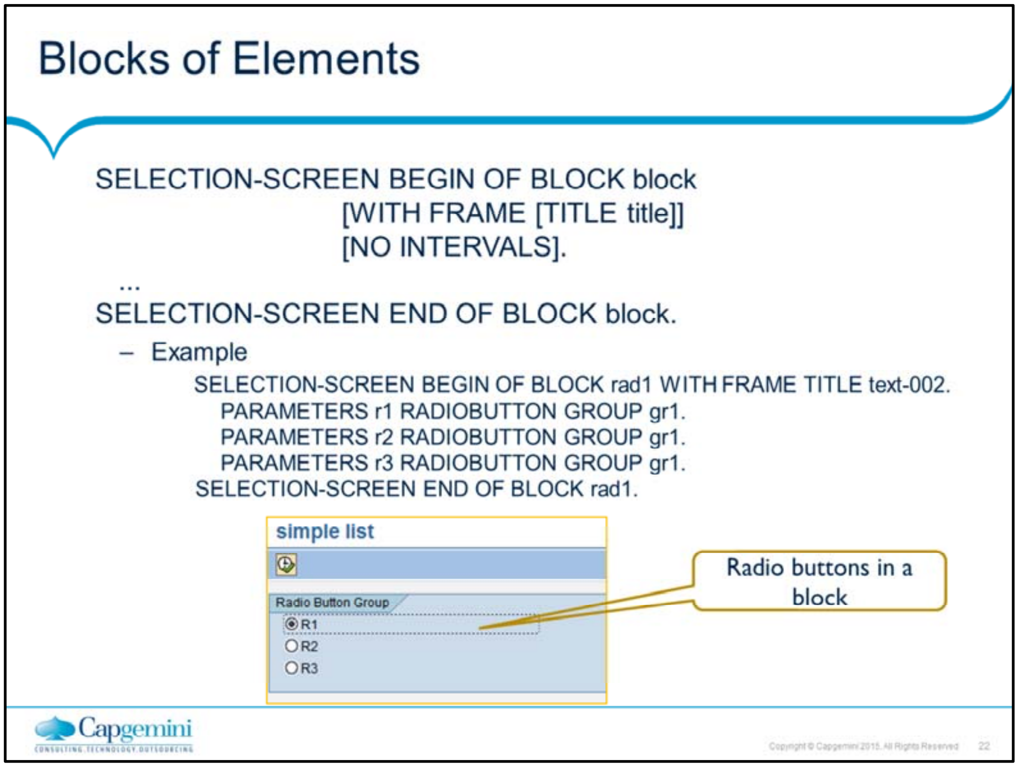
```
SELECTION-SCREEN BEGIN OF LINE.
SELECTION-SCREEN COMMENT 1(10) text-001.
PARAMETERS: p1(3) TYPE c, p2(5) TYPE c, p3(1) TYPE c.
SELECTION-SCREEN END OF LINE.
```

Program Edit Goto System Help
simple list
Name

Parameters :P1, P2 and P3 in single line

Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

selection-screen begin of line and **selection - screen end of line**-All input fields defined between these two statements are placed next to each other on the same line.



Blocks of Elements

```
SELECTION-SCREEN BEGIN OF BLOCK block
                    [WITH FRAME [TITLE title]]
                    [NO INTERVALS].
  ...
SELECTION-SCREEN END OF BLOCK block.
```

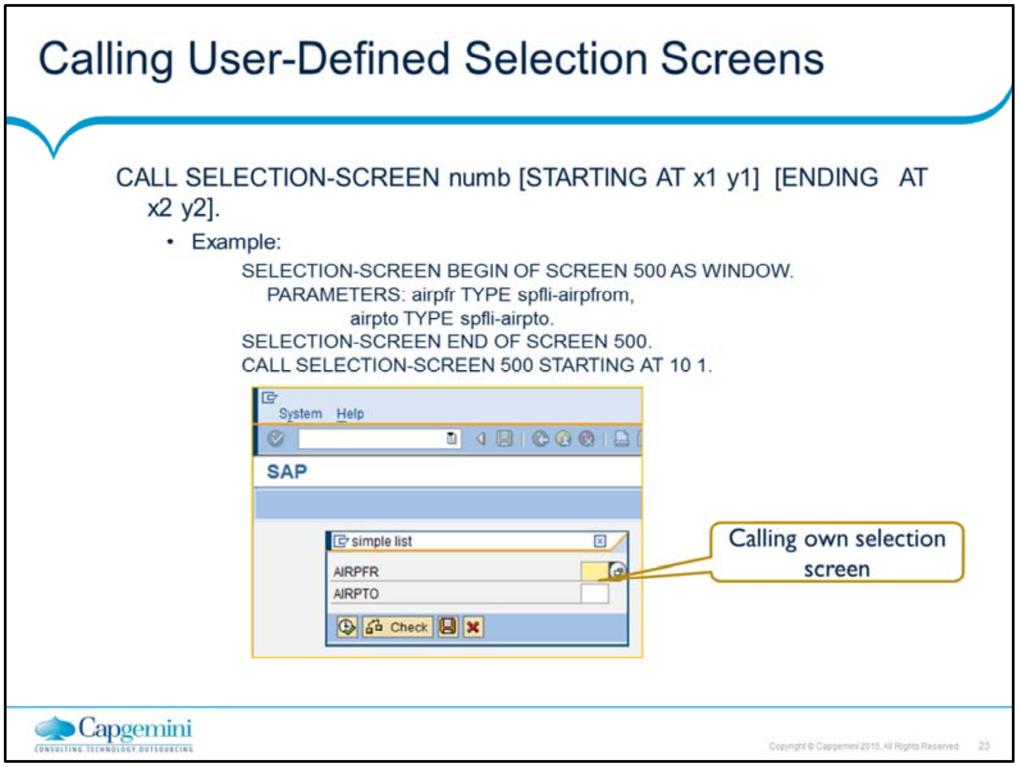
- Example

```
SELECTION-SCREEN BEGIN OF BLOCK rad1 WITH FRAME TITLE text-002.
  PARAMETERS r1 RADIOBUTTON GROUP gr1.
  PARAMETERS r2 RADIOBUTTON GROUP gr1.
  PARAMETERS r3 RADIOBUTTON GROUP gr1.
SELECTION-SCREEN END OF BLOCK rad1.
```

Radio buttons in a block

Selection screen elements can be combined into cohesive units called blocks. These logical blocks are, in essence, a cosmetic screen feature that encapsulates a combination of screen input elements and can be created with a descriptive frame title.

Logical blocks help to make the selection options easier to understand and use.



Calling User-Defined Selection Screens

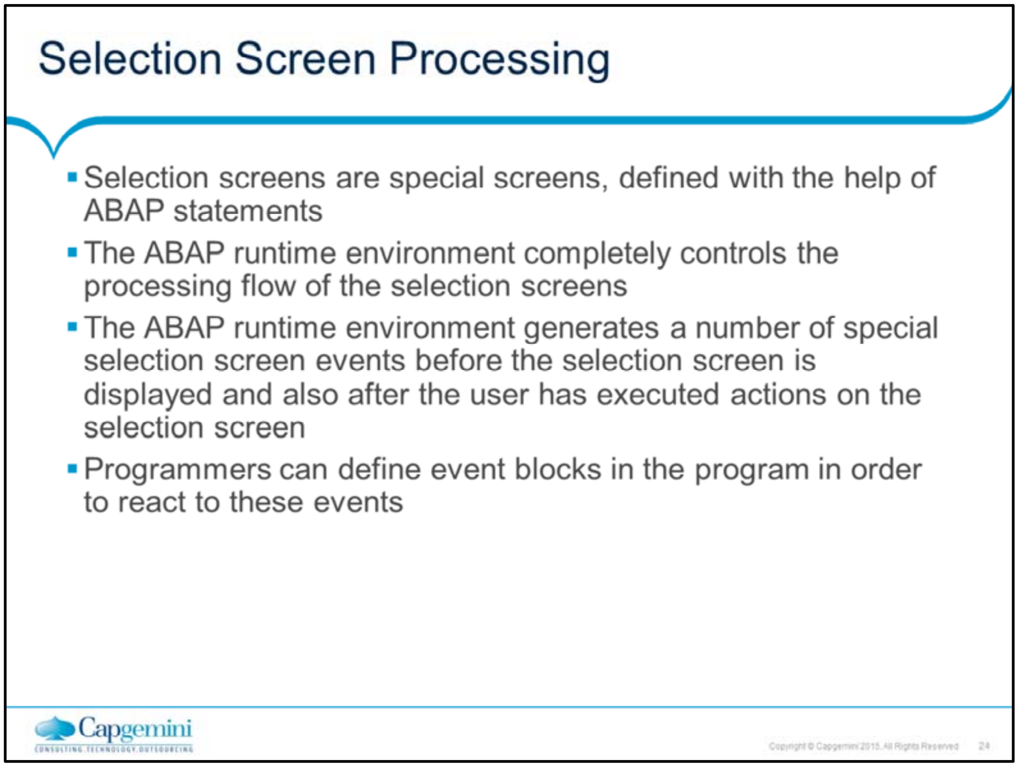
CALL SELECTION-SCREEN numb [STARTING AT x1 y1] [ENDING AT x2 y2].

- Example:

```
SELECTION-SCREEN BEGIN OF SCREEN 500 AS WINDOW.
  PARAMETERS: airpfr TYPE spfli-airpfrom,
          airpto TYPE spfli-airpto.
SELECTION-SCREEN END OF SCREEN 500.
CALL SELECTION-SCREEN 500 STARTING AT 10 1.
```

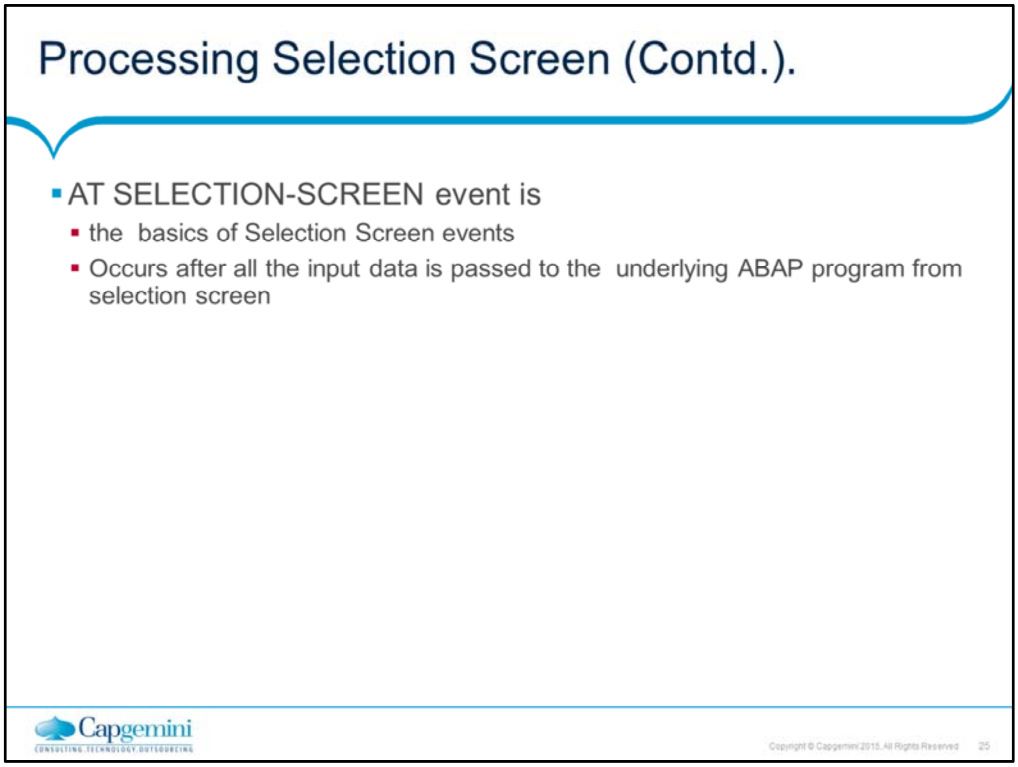
Calling own selection screen

23



Selection Screen Processing

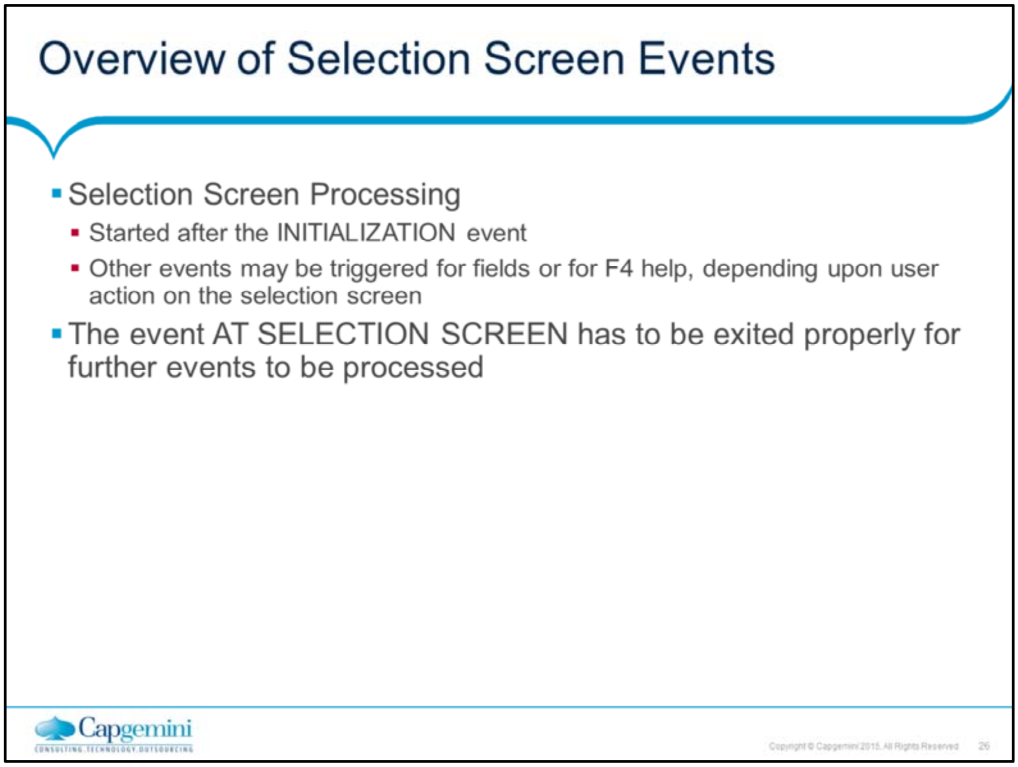
- Selection screens are special screens, defined with the help of ABAP statements
- The ABAP runtime environment completely controls the processing flow of the selection screens
- The ABAP runtime environment generates a number of special selection screen events before the selection screen is displayed and also after the user has executed actions on the selection screen
- Programmers can define event blocks in the program in order to react to these events



Processing Selection Screen (Contd.).

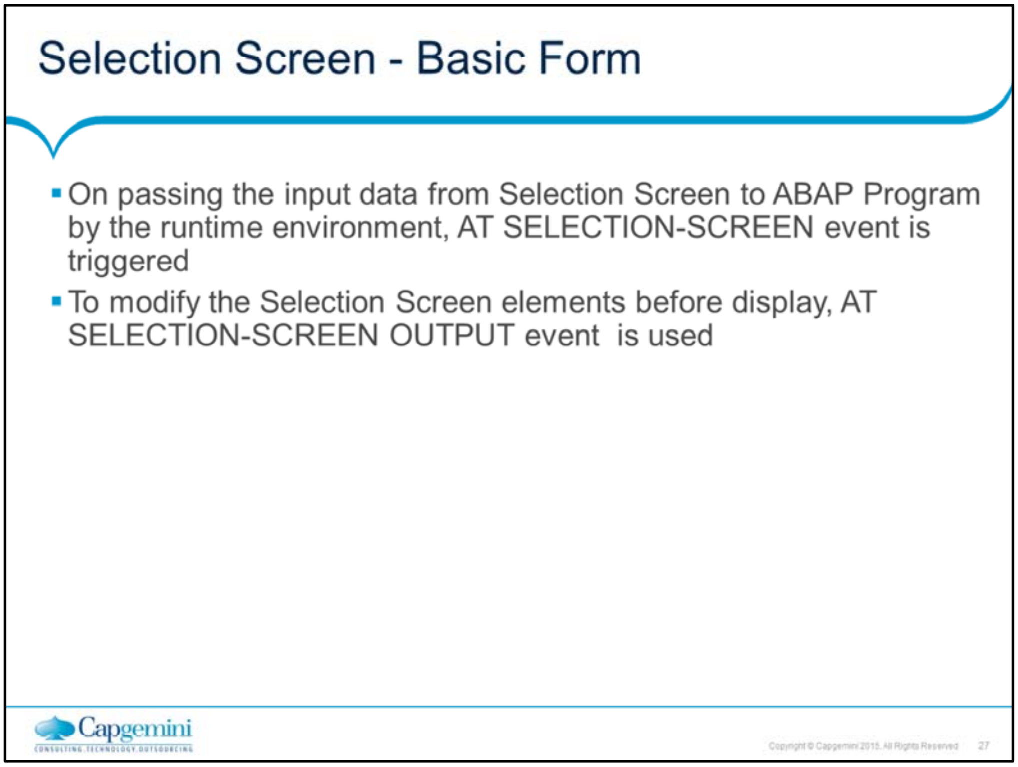
- AT SELECTION-SCREEN event is
  - the basics of Selection Screen events
  - Occurs after all the input data is passed to the underlying ABAP program from selection screen

25



Overview of Selection Screen Events

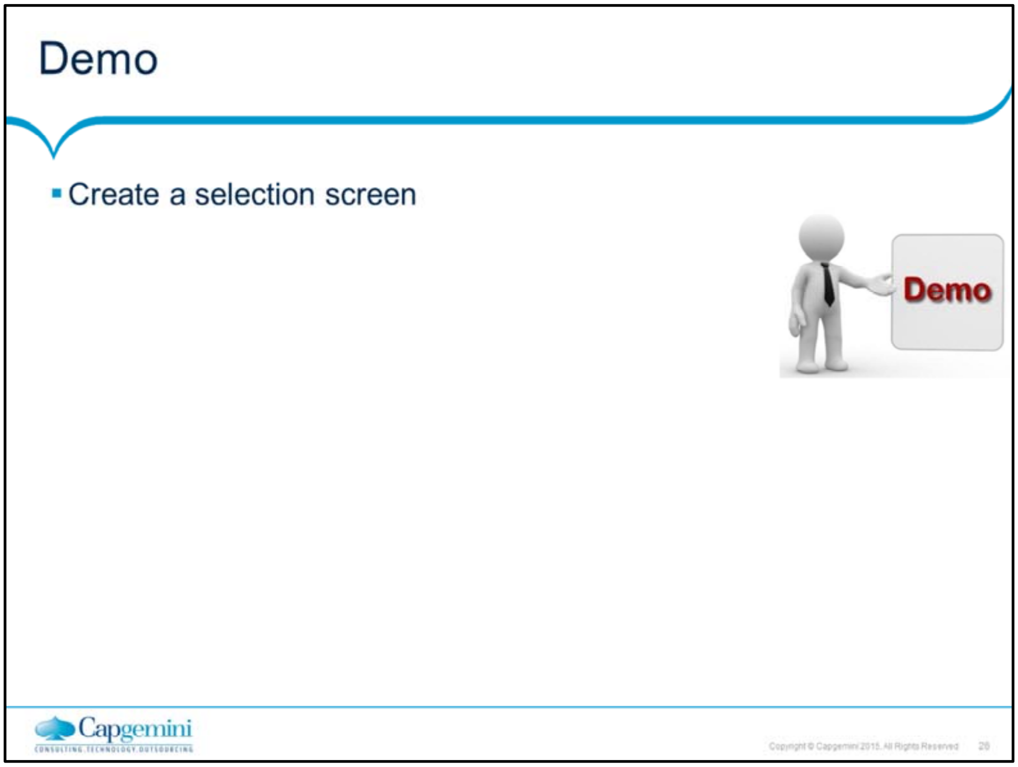
- Selection Screen Processing
  - Started after the INITIALIZATION event
  - Other events may be triggered for fields or for F4 help, depending upon user action on the selection screen
- The event AT SELECTION SCREEN has to be exited properly for further events to be processed



Selection Screen - Basic Form

- On passing the input data from Selection Screen to ABAP Program by the runtime environment, AT SELECTION-SCREEN event is triggered
- To modify the Selection Screen elements before display, AT SELECTION-SCREEN OUTPUT event is used

27

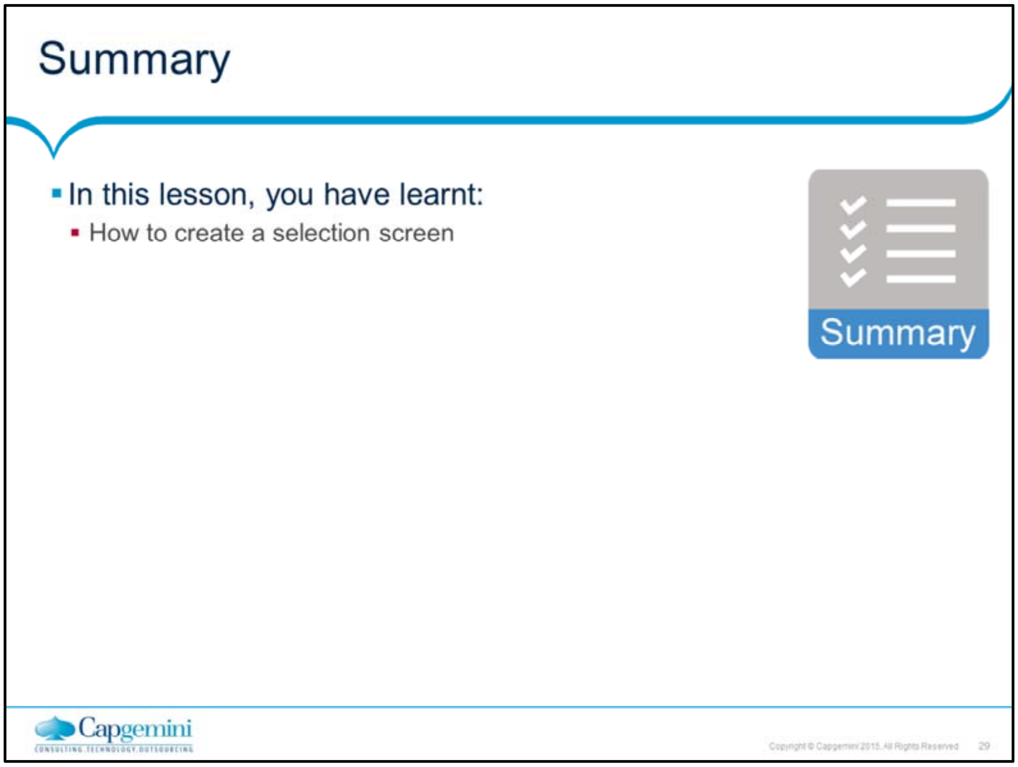


Demo

- Create a selection screen

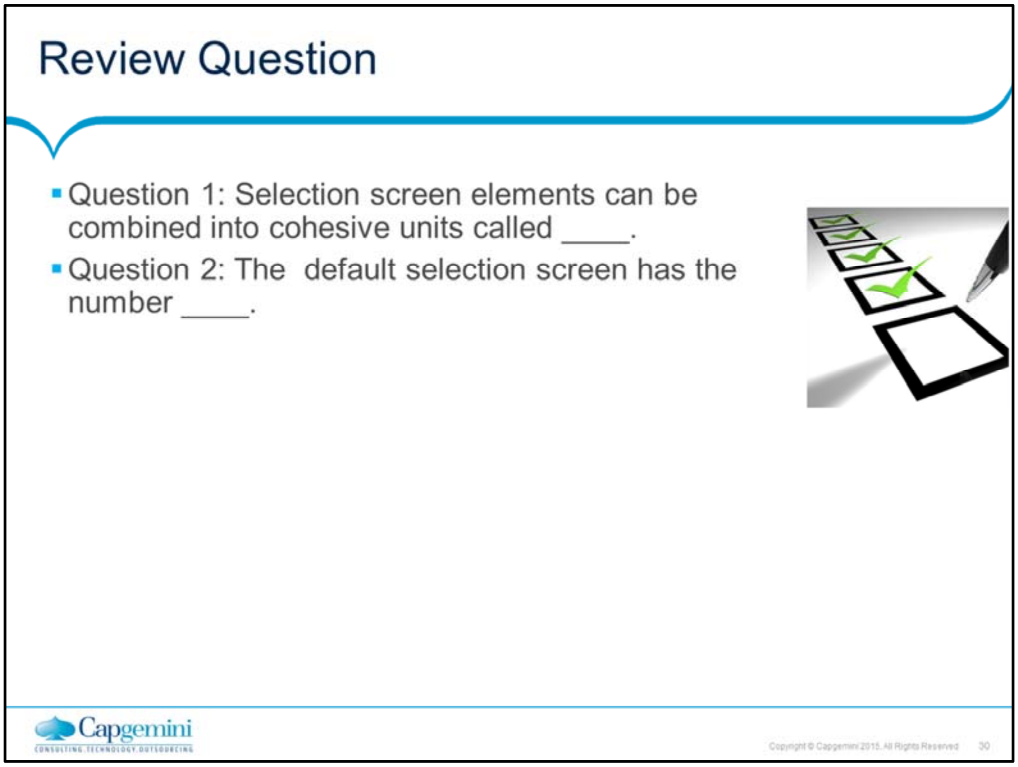
Capgemini
CONSULTING.TECHNOLOGY.OUTSOURCING

28



Summary

- In this lesson, you have learnt:
  - How to create a selection screen



Review Question

- Question 1: Selection screen elements can be combined into cohesive units called ____.
- Question 2: The default selection screen has the number ____.