UNIT-2

CHAPTER 3: JAVA GUI PROGRAMMING REVISION TOUR-I

Rapid Application Development: It describes a method of developing software through the use of pre-programmed tools or wizards. The pre-programmed tools or controls are simply dropped on a screen to visually design the interface of application. It enables program development in shorter time.

NetBeans Java IDE: It is a free, open-source, cross-platform IDE with built-in support for Java programming language. It has more advanced GUI building tools available in any open-source Java IDE.

Event: Occurrence of an activity.

Message: Information sent to the application or received from the application.

Types of Swing Components:

(a) Component: It is a self-contained graphic entity like JLabel, JTextField etc.

(b) Container: It can hold other components. It is of two types:

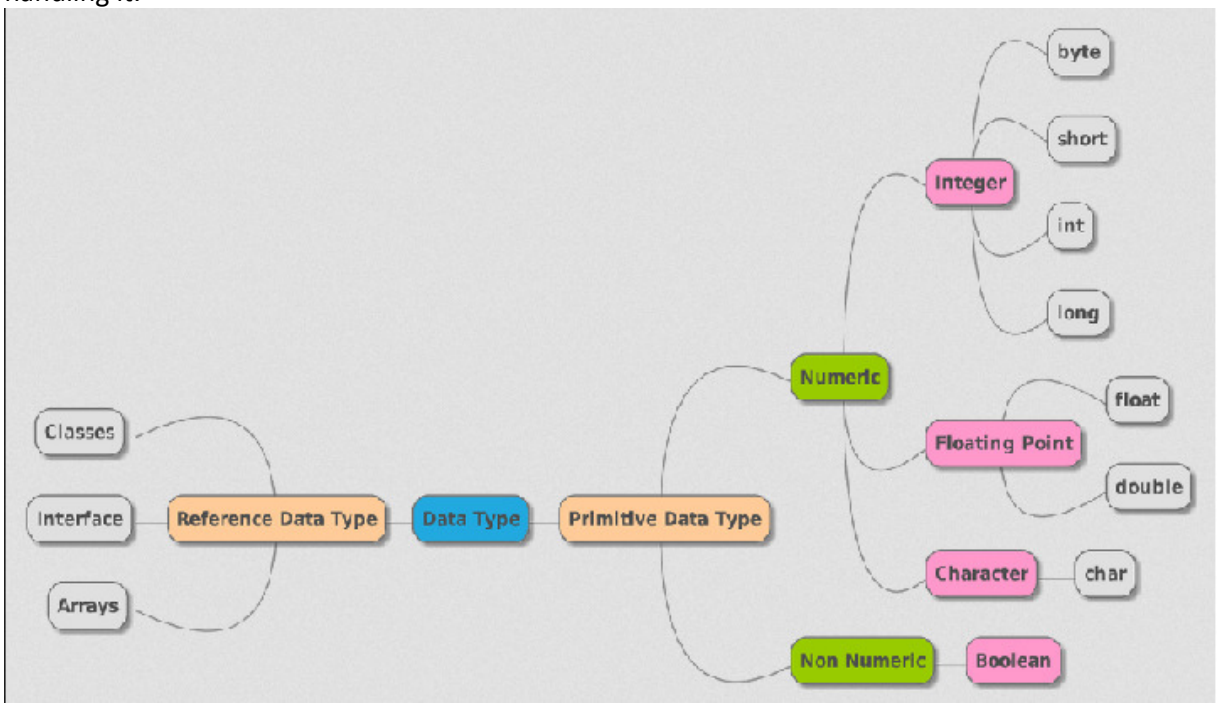
(i) Top Level Container: Can be displayed directly on a desktop. Every swing application must have at least one top level container, i.e. JApplet, JDialog, JFrame.

(ii) Non Top Level Container: Can be displayed within another top level container, i.e. JPanel, JScrollPane, JInternalFrame, JLayeredPane etc.

Child controls are controls inside a container control.

Java Character Set: It is the set of valid character set that a language can recognise. Java uses *Unicode character set*.

Data Types: Data Types are means to identify the type of data and associated operations of handling it.



Variables: It is a named memory location which holds a data value of a particular data type.
Variable declaration & Initialisation:
A variable with declared first value is said to be initialized variable. e.g. int rollno = 1;

Text Interaction:

(i) getText() Method : String name = nametf.getText();
(ii) setText() method: Stores or changes text in text based controls e.g. ranktf.setText("1");
(iii) Parse() Methods: Used to convert a string value into numeric value.
(iv) JOptionPane.showMessageDialog (): Displays a Message Box.

System.out.print (): It prints the desired output on output window.
System.out.println(): It prints the desired output on output window and also changes the line.

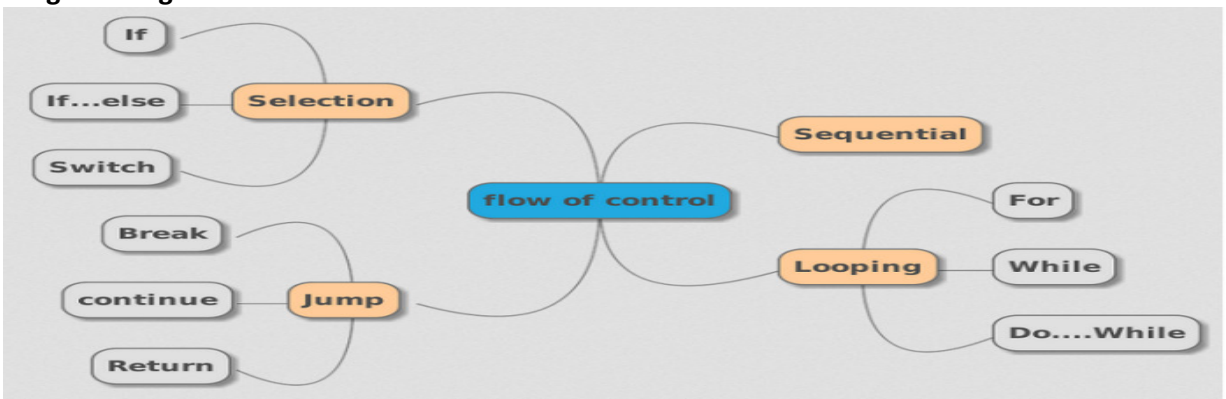
Variable Scope: Program region within which a variable is accessible. It is accessible within the set of braces it is declared in.

```
if (..)
 {
int x = 5;
....
}
System.out.println("The result is :" + x);     //error
```

Constant: Named memory locations whose value can't be changed in the program. It makes program easy to read, check & modify. e.g. final int rateofinterest = 10;
Operator: It represents particular task and the objects of the operation is called operands. It may be Unary (working on one operand, eg. unary -), binary (working on two operands, e.g. subtract -) or ternary (working on three operands, e.g. ?:).
Type Conversion: It is the process of converting one predefined data type into another. It has two forms:

- **Implicit (Coercion):** Compiler automatically performs conversion to the type of the largest operand in mixed mode expression.
- **Explicit (Type casting):** Done by programmer. Converting to or from Boolean is not permitted. : (type) expression e.g. (float) (x/5 * y + 5)

Programming Constructs: Flow of Control



Selection:

If statement;
if (expression)
statement ;
if . . . else statement
if (expression) statement 1 ;
else statement 2;
Nested if:
if (expression) {
if (expression)
statement 1 ;
[else
statement 2;] }
else
body of else;
	if else if
	if (expression)
	body ;
	else if (expression)
	body;
	.
	else body;
	Dangling else Problem:
	if (expression)
	if (expression) statement 1 ;
	else statement 2;
	(Indentation indicates else for outer if but it actually matches with preceding If.)

The Switch Statement:

```
switch (expression)
{       case constant 1 : statement sequence 1;
                          break;        //  value of expression must be byte, short, int or a char.
        case constant 2 : statement sequence 2;
                          break;
        .
        [default : statement sequence n];          //default can be anywhere need not to be in last.
}
```

In absence of break in switch statement, Java will start executing all statements after the correct case. This phenomenon is called **Fall-Through.**

Iteration (Looping) statements:

(i) for loop (ii) while loop (iii) do-while loop

Parts of loop:

- Initialisation Test Expression Update Expression Loop Body

for loop:

```
for (int i = 1; i <= 10; ++i)
System.out.print( i + " ");
```

Program to find factorial using for loop:

```
int fact = 1, a;
int num = Integer.parseInt(numtf.getText());
for (a = 1; a <= num; a++){
fact = fact*a;      }
System.out.println("The factorial of " + num + " is " + fact);
```

The While Loop: It is an Entry-controlled/Top-tested/Pre-Tested loop.

Program to find factorial using while loop:

```
int num = Integer.parseInt(numtf.getText());
long i = num, fact = 1;
while (num != 0)  {
fact = fact *num;
-- num; }
System.out.println("The factorial of " + i + " is " + fact);
```

The do-while Loop: It is an Exit-controlled/Bottom-Tested/Post-Tested loop. It executes at least once.

Syntax:

```
do {statement ;
    } while (test-expression);
```

Example:

```
char ch = 'A';
do {System.out.println(ch);
    ch++;
} while (ch <= 'Z');
```

Jump statement

- **Return:** To return from a function.
- **Break:** It is used to terminate the body of while, do-while, for or switch statement and transfers control to the statement following the body.
- **Continue:** It skips the rest of the loop (while, do-while or for) and starts the next iteration.

Solved Questions:

1. Write a java code to find out whether a year (4 digit number stored in a variable) is a leap year.

```
private void lybutActionPerformed(java.awt.event.ActionEvent evt) {
long i = Long.parseLong(lytf.getText());
if (i%100 ==0)
if (i%400 ==0)
    System.out.println(i +  " is Leap Year");
else        System.out.println(i + " is Normal Century Year");
else if (i%4==0)
System.out.println(i + " is Leap Year");
else        System.out.println(i + " is not a Leap Year"); }
```

2. Write a java program to find the greatest out of three numbers.

```
int a = Integer.parseInt(tf1.getText()), b = Integer.parseInt(tf2.getText());
int c = Integer.parseInt(tf3.getText());
if (a>b && a>c)
System.out.println( a + " is greater");
elseif (b>c)        System.out.println( b + " is greater");
else                    System.out.println( c + " is greater");
```

Sample Questions:

1. What will be content of the jTextArea1 after executing the following code:
   for (int i=2;i<=5;i++)
   { jTextArea1.setText(jTextArea1.getText()+" " + Integer.toString(i*i)); }
2. Write java code that takes value for side of a square in jTextField1 and calculate area of it which is to be displayed in jTextField2.
3. Item code consisting of 5 digits is stored in a integer type variable intItemCode. Write the code for keeping this itemcode in a String type variable called strItemCode.
4. What message will be displayed after the execution of the following code?
   int age =64, relaxation = 4; modiage= age – relaxation;
   if (modiage < 60) jOptionPane.showMessageDialog(null, "Not Eligible");
   else jOptionPane.showMessageDialog(null, "Eligible");
5. Rewrite the following program code using if statement:
   int c = jComboBox1.getSelectedIndex();
   switch(c)
   { case 0 : amount = bill; break;
   case 1 : amount = 0.9 * bill; break;
   case 2 : amount = 0.8 * bill; break;
   default : amount = bill;
   }