Akshatha.V.L
4AL18EC005

10/08/2020

- Copper Pour: Place a copper poar go to Add filled zones and drawn within the boundaries on the silk screen and select the layer. we want front layer. and it as to be associated with GND.
  And drawn ~~rect~~ boundary within the circuit board.
  To fill right click and select fill the zones.
  To cop with botton side change F.Cu to B.Cu and copper Pour & with ground. we got the copper Pour on botton side & front side.
  we write the Text so select T and select Foul text silk screen. and write the text on silk screen.
  This is 2 layer design we may have 4 or 6 layer design.

- Mounting holes to PCB:
  first remove copper Pour or fill before mounting holes or components.
  1 Pin, preferencence - General - Inches / Millimeters / choose millimeters. 3 mm as to be taken., diameter - 5 mm, hide the name of silk screen. After that enable the copper fill then after placing the 1 Pin on both the side we are done with mounting holes to PCB. Same alterations has to be done with the mounting

10/06/2020

Akshatha.Y.C
4AU18EC005

→ Part 1

① Creating a Hello World Java Program.

Workbench → File → New Java project → Project name → Nxt → Finish → Tutorial → Right click → New → class → Name it. and click on public void main → Finish.

```
public class Application {
/**
* @ param args
*/
public_ static world main (string[] args) {
    // TODO Auto-generated method stub.
}
}.
    System.out.println("Hello World");
        Then run it
```

→ Using Variables:

```
public class Application {
    public static void main(String[] args) {
        int myNumber; (declaration)
        myNumber = 88; (initialization)
        System.out.println(myNumber);
    }
}
```

Here declaration & initialization cannot be done in 2 different steps

So

```
int myNumber = 88;
```

→ There are 8 primitive types of variables in Java.

```
public class Application {
    public static void main(String[] args) {
        int myNumber = 88;
        short myshort = 847;
        long mylong = 9797;
        double myDouble = 7.3243;
        System.out.println(myNumber);
        System.out.println(myshort);
        System.out.println(mylong);
    }
}
```

```
float myfloat = 324.3f;
char mychar = 'y';
boolean myBoolean = true;
Byte mybyte = 127;
```

- Strings: Working with text:

Here will study about non primitive types:

new project → Name → Next → Finish.

Tutorial → New → class → Here we take Inherited abstract method → Finish

String is a class + type of object which can hold variable.

```
public class Application{
    public static void main (String[] args){
    int myInt=7;
    String text= "Hello";
    System.out.println(text);
    }
}
```

→ while loops:
```
int value=0
while(value<10)
    {
    System.out.println("Hello"+value");
    value = value+1;
    }
```
control+shift+F

→ For loops:
```
for(int=0; i< 5; i++){
    System.out.println("Hello");
    }
```
↳ "The value of i is: %d %d\n to next line", i);

→ "if":
```
if(4==4){
    System.out.println("Yes, it's true!");
```
oo.
we can use variable.
```
int myInt = 20
if(myInt<10){
    System.out.println("true")
}else{ or else if (myInt > 20){
    System.out.println("false")
}else{
    System.out.println("None of the above");
}
```

- Getting User Input:
```
import java.util.scanner
scanner i/p= new Scanner(System.in);
System.out.println("Enter a line of text:");
String line= i/p.nextline();
System.out.println("You entered: "+line);
```

√5.33 + 10

→ Do... while:

```
int value= scanner. nextInt();
while (value ! = 5){
   system.out .println("Enter a number:");
   value = Scanner . nextInt();
}
```

→ Switch:

```
scanner input= new scanner(system.in);
system.out. println(" Please enter a command:");
string text= input. nextline();
switch (text){
case "start":
 system.out. println ("Machine started");
 break;
case "stop":
   System.out. println("Machine stopped");
   break;
 default:
   system.out. println("Command not regonized");
 }
```

→ Arrays:

```
int value= 7;
int[] value;
values= new int[3]
System.out.println (values[0]);
values [0]=10 ;
values [1] = 20;
values[2] = 30;
system.out println (values(0))
  "      "     "    (values(1) )
```