11/06/2020

Date______ Page______

Akshatha.Y.E
4AL18EC005

• Arrays of Strings:

```
public static void main (string[] args){
 string[]words = new string [3];
  words [0] = "Hello";
  words[1]  = "to";
  words [2] = "you";
System.out.println(words[2]);
String[] fruits = {"apple, "banana", pear", "kiwi"}
 for (string fruit: fruits){
      System.out.println(fruit);
}
  int value = 0;
  string text = null;
 System.out.Println(text);
 string[] texts = new string[2];
 system.out.println(texts[0])
   texts[0] = "one";
 }
```

• Multidimentional Array:

```
int[]values = {3,5, 2343};              } 1 dimensional
System.out.println (values [2]);        }   array
int[][] grid = {
          {3, 5, 2343},
          {2, 4},                       } Multidimentional
          {1, 2, 3, 4}                  }   array
 }
  System.out println(grid [1][1]);
  system.out.println(grid [0][2]);
for(int row=0; row<grid.length; row++){
 for(int col=0; col<grid[row].length; col++){
 System.out.print(grid[row][col] + "\t");
}
 system.out.println();
}

String [][] words = new string[2][1]
System.out.println(words[0]);
words[0] = new string[3];
words[0][1] = "hi there";
System.out.println(words[0][1]);
 }
 }
}
```

- **Classes and objects:**

public class Redfruit{

class: is a template or blueprint to create a object. Eg: class Person{

Classes can contain data, subroutines (main) (methods), Instance variables (data or "State");

String name; int age

```
        person person1 = new Person();
        person1.name = "Joe Bloggs";
        Person1.age = 37 ;
        Person person2 = new Person();
        person2.name = "Sarah Smith";
        Person2.age = 20;
    System.out.println(person1.name);
```

- **Methods:**

```
    void speak(){
        System.out.println ("Hello");
```

To import with variable

```
        person1.speak();
        person2.speak();
  for (int i=0; i<3; i++){
    System.out.println("My name is:"+name+"
and I an"+age+" "+"years old");
   }
  }
    void sayHello(){
       System.out.println ("Hello there!");
    }
```

To execute

```
      Person1.sayHello();
```

- **Getters and Return Values:**

```
        return yearsleft;
   }
   int getAge(){
        return age;
    }
  }
  public class App{
     while concluding
       int age = person1. getAge();
     }
   }
```

- **Method parameters:**

Parameter is like a variable u pass in to a method.

```
      class Robot{
            public void speak (string text){
            System. out. println(text);
        public void jump (int height){
            System.out. println ("Jumping: "+height);
  public void move(string direction, double distance){
     system . out. println("Moving "+ distance+
      "metres in direction"+ direction);
     }
     }
     sam. speak ("Hi I'm sam")
     sam. jump(7);
     sam. move ("West", 12.2);
     string greeting = "Hello there"; ______ string
```

- **Setters and this:**

```
          frog frog1 = new Frog()
          frog1. name = Bertie";
           frog1. age = 1;
```

```
System.out. println (frog1.getName());
```

```
this name = name;  } " this "
this .age = age;   }
 return name;
 return age;
```

- **Constructors:**

```
        class Machine {.
```

Constructor is a special method to run the machine.

```
        Machine.machine1 = new Machine ();
    public Machine (string. name) {
        System.out.println ("Seconds constructor
                running }
        this. name = name;
```

- **Static:**

```
class Thing {
        public final static int Lucky Numbers;
        public string name ;
        public static string description;
        public void showName () {.
        public static void show Info() {.
```

- **String Builder & string Formatting;**

```
    String builder s = new string Builder.
        s. append ("My name is Roger.")
        .append (" ")
        .append (" I an. a skydiver.")
    System.out.println(s.tostring());

  for (int i=0; i<20; i++) {
      System.out.printf ("%-2i; %.s\n", i,
                        Here is some text");
```

11/08/2020

Akshatha.Y.G
4AL18EC005.

Date
Page

KI-CAED[PCB]

5. Create a library and put your own component in that library.

→ First search where the Kicad save the file. First Explore → C-drive - prog - 6 - KiCad. Share - Library - Default Kicad library. When update comes in Kicad so the files may charge so it has to be saved. Private - Kicad project - My libraries. Kicad - Schmatics. - Only folder - No library - we have to create a library, so export the component to the library. Bring component - add simple component. Delete the Body of resistor make it American version as —[R]— to —/\/\/— Kicad → My Libraries - Name it (My Library) Preference - Library → Use defined search path. → Add → Drop box - Libraries → Choose folder. → No → Add library → My library - R is found. Again → Prefreences → R (component) → My Library → OK. →

- Create a Custom footprint component:
  Kicad → PCB drawing → Create new footprint → New module → Name of component footprint MySOICn → OK, E - Edit - value** to U** asters are replaced by U¹ and U², place one pad type SMD, rectagular on origin, keep X_1 = 0.60mm. Y_1 = 1.55mm., E = 1.27mm, whenever we move it spaces to 1.27mm. we need 4 pads, C = 5.40mm. If it not there is grid size we have to go for user defined. selectively.
  we need 6 pads