**Q: HOW TO CREATE PACKAGES AND WHAT IS BEST WAY TO GIVE NAME**

Packages in java is to combine group of classes, interfaces and sub packages.Predefined packages are present in java. Like java. Lang, java.io, java. util etc. to perform basic operations.

**WHAT IS MAIN METHOD WILL DO?**

Main method will resemble the logic of a program.

**creating property/data members :**

**Q: what is data type and different data types`**

Data types represent type of variables and decide memory size and range.

Data types are of 8 types. They are:

|  |  |
| --- | --- |
| **Data types** | **Size** |
| Byte | 1 byte |
| Short | 2 bytes |
| Int | 4 bytes |
| long | 8 bytes |
| Float | 4 bytes |
| double | 8 bytes |
| char | Single Space |
| Boolean | True or False |

**Q:** **What is variable?**

**variables** are of 3 different types. They are

1) Local variables

2) Instance variables

3)Static variable

**Local Variable:**

1. Local Variables means variables which are mentioned inside the method. Scope of these local variables lies within the method or blocks. Memory allocation for this Local variables starts when method starts and memory destroyed when method ends.
2. Initial values are not assigned.
3. Local variables inside the method can directly accessed.
4. Stack memory is used as stored memory for local variables.

**Instance variable:**

1. Instance variables are located inside the class outside the method. Scope lies within the class. Memory is created when object is created and memory ends when object is terminated.
2. Object is created when instance variables need to access in different areas.
3. Heap memory is used for memory storage.
4. Variables need to initialize.

**Static Variables:**

1. Static variables are located outside the method with static modifier. Scope of variables to access within the class.
2. Memory allocated when .class files are loaded and memory destroyed when .class files are unloaded.
3. Non Heap Memory is used for memory storage.

**Q: creating method with void(27,60)**

Class Test

{

void m1()

{

System.out.println(“M1 Method”);

}

public static void main (String [] args)

{

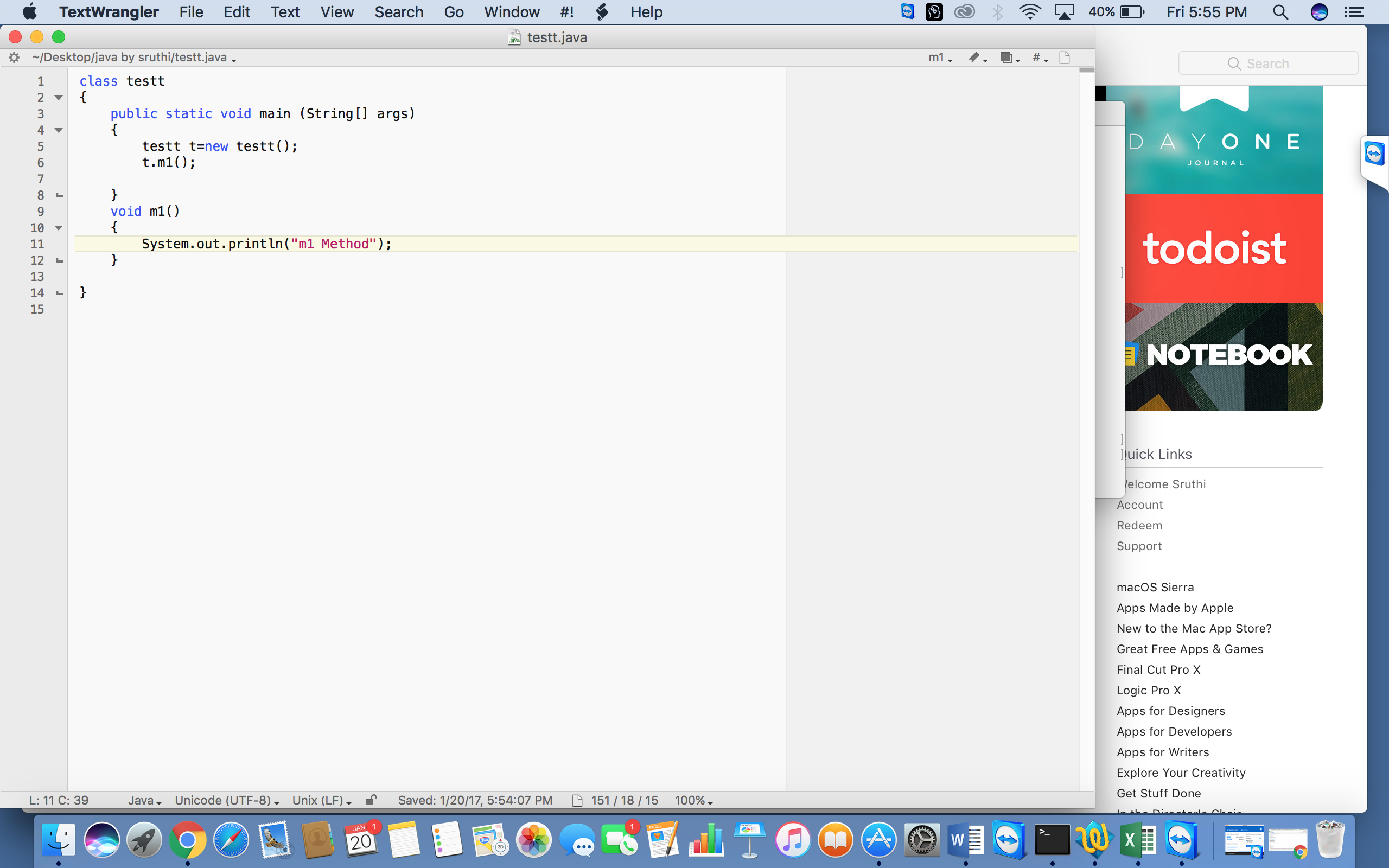
Test t=new Testt ();

t.m1();

}}

output:

M1 Method.



**Q: creating variable, we can create variables inside method(30)**

Variables can create inside the method and those methods are called Local variables.

Program:

Class Test

{

public Static void main (Strings [] args)

{

int a=10;

int b=12;

System.out.println(a+b);

}

}

**Q: creating method with return data type, we can return int/string/double/float/date etc(method that will return hard coded value)**

Class Test

{

int a=10;

int b=20;

int c=30;

int add()

{

int d=a+b+c;

return store d;

}

Public static void main(string[] args)

{

Test t=new Test();

Int e= t.add();

System.out.println(e);

}

}

Output:

60

**Q) (OPT)Method that will return hard coded value**

class Tesst

{

int a=10;

void m1()

{

System.out.println("m1 method");

}

public static void main(Strings[] args)

{

Tesst t =new tesst();

t.m1('a');

}

**Q) Create default/paramterzied constructors**

Class Test

{

Test()

{

system.out.println(“0-arg constructor”);

}

Test(int a)

{

System.out.println(“1-arg constructor”);

}

public static void main(String[] args)

{

new Test();

new Test(88);

}

}

**Q) Method that will return property value?**

**Q)** **Creating method with return data type and parameter?**

calling method with return data type and parameter:

Class Test

{

int a=10;

int b=20;

static int c=30;

static d=40;

void m1()

{

System.out.println(a);

System.out.println(b);

System.out.println(Test.c);

System.out.println(Test.d);

}

static void m2()

{

Test t=new Test ();

System.out.println(t.a);

System.out.println(t.b);

System.out.println(Test.c);

System.out.println(Test.d);

}

public static void main(strings[] args)

{

Test t=new Test ();

t.m1();

Test.m2()

}

}

output:

10

20

30

40

10

20

30

40

**Q: CREATING STATIC METHOD?(58,50)**

Class Test

{

Static int a=10;

Static int b=15;

Public static void main(string[] args)

{

Test t = new Test ()

System.out.println(t.a);

System.out.println(t.b);

}

}

**Q: CREATING STATIC BLOCK?**

Static block is mostly used for changing the default values of static variables. This block gets executed when the class is loaded in the memory. A class can have multiple Static blocks, which will execute in the same sequence in which they have been written into the program.

**Q: Creating object?**

The difference is, the default constructor is public without the keyword static, and has no return type. Meaning, it does not have void, or int, or anything else. Also, the name of the constructor is the same as the name of the object. all constructors must be the name of the object

**Q: Calling method with void?**

class Test

{

void m1()

{

System.out.println(“m1 method”);

}

}

**Q: Calling method with no return and parameter?**

Class Test

{

void m1(int a,int b)

{

system.out.println(a);

system.out.println(b);

}

void m2(float c,char d)

{

system.out.println(c);

system.out.println(d);

}

public static void main(String[] args)

{

Test t=new Test();

t.m1(10,20);

t,m2(10,8f,’r’);

}

}

**Q: Calling method with return and no parameter?**

Class Test

{

float diff()

{

float a=11.8f;

float b=22.5f;

float c=a-b;

return c;

}

public static void main(String[] args)

{

Test t=new Test();

Float d= t.diff();

System.out.println(d);

}

}

**Q: Calling method with return and parameter?**

Class Test

{

int a=11;

int b=12;

void m1(int a,String str);

{

System.out.println(a);

System.out.println(Str);

}

int add()

{

int c=a+b;

return c;

}

void m2(float f,double d)

{

System.out.println(f);

System.out.println(d);

}

public static void main(string[] args)

{

Test t=new Test();

t.m1(10,”Sruthi”);

t.m2(10.5f,12.5);

int e=t.add();

System.out.println(e);

}

}

**Q: Calling method with return and storing the return data?**

**Q: Calling static method?**

Class Test

{

int a=100;

int b=200;

static void m1()

{

Test t=new Test();

System.out.println(t.a);

System.out.println(t.b);

}

static void m2()

{

Test t=new Test();

System.out.println(t.a);

System.out.println(t.b);

}

Public static void main(String[] args)

{

Test.m1();

Test.m2();

}

}

**Q:** **What is final keyword, create final class, final method, final property?**

The final keyword in java is used to restrict the user. The java final keyword can be used in many context. Final can be:

1. variable
2. method
3. class

The final keyword can be applied with the variables, a final variable that have no value it is called blank final variable or uninitialized final variable. It can be initialized in the constructor only. The blank final variable can be static also which will be initialized in the static block only. We will have detailed learning of these. Let's first learn the basics of final keyword.

**Q: write code for creating abstract class**?

Class test

{

**Q: Implement method overloading?**

Class Test

{

void m1(int a)

{

system.out.println(“int m1 method”);

}

void m1(int a,int b);

{

System.out.println(“int,int m1 method”);

}

void m1(char a)

{

System.out.println(“char m1 method”);

}

public static void main(String[] args)

{

Test t= new Test();

t.m1(10);

t.m1(10,20);

t.m1(‘r’);

}

}

**Q: Implement method overriding?**

Class parent

{

parent()

{

system.out.println(“ parent class 0-arg constructor”);

}

}

class child extends parent

{

child()

{

super();

system.out.println(“child class 0-arg constructor”);}

child (int b)

{

system.out.println(“child class 1-arg constructor”);}

public static void main(string[] args)

{

new child();

}  
}

37,39,42,45,47,52,62,64

**66Answer:/68/70**

**72 ques---74 answer**

dxasdsde

37)

class tess

{

  void sum (int a,int b)

  {

    System.out.println(a+b);

  }

  void sum( int a,int b,int cs

  {

    System.out.println (a+b+c);

  }

  public static void main (String[ ]args)

{

  tess n=new tess();

  n.sum (10,10,10);

  n.sum (20,20);

  }

}