1. What is variable?
2. Variable is a name of memory location. It will store the value in memory
3. There are three types of variables in java: local, instance and static.

#### Local Variable- A variable which is declared inside the method is called local variable.

#### Instance Variable- A variable which is declared inside the class but outside the method, is called instance variable. It is not declared as static.

#### Static variable- A variable that is declared as static is called static variable. It cannot be local.

Example:

**class** A{

**int** data=50;//instance variable

**static** **int** m=100;//static variable

**void** method(){

**int** n=90;//local variable

}

}//end of class

To create variable we specify

Variablename datatype;

2)What is data type and different data types`

Data types represent the different values to be stored in the variable. In java, there are two types of data types:

1. Primitive data types – Primitive datatypes are predefined by the language and named by a keyword.

Types: int, short, long, double, float, byte, boolean, char

1. Non-primitive data types **are** not defined by the programming language, but are instead created by the programmer. They are sometimes called "reference variables," or "object references," since they reference a memory location, which stores the **data**.

creating property/data members : we create properties at class level

int salary

String firstname

creating method with void : we write methods in

Ex: Void MethodName();

creating method with void and parameter

Ex: Void MethodName(int param1);

creating method with return data type

Ex: int MethodName();

     String MethodName();

creating method with return data type and parameter

int MethodName(int id) {

Int id;

Return id;

}

String MethodName();

creating variable

Int var1;

creating static property: All instances shared the value http://crunchify.com/java-static-methods-variables-static-block-and-class-with-example/

creating static method

http://crunchify.com/java-static-methods-variables-static-block-and-class-with-example/

3). creating object:

An object is an instance of a class. It has state and behavior and stores entire class information.

<classname> <objectname> = new <classname>();

13. calling method with no return and no parameter

Ex:

package examplepack;

public class Mathematics

{

Void sum() /\* No arg \*/

{ int a=10;

int b=20;

int c=a+b;

System.out.println(“addition value is:” +c);

}

public static void main(String[] args)

{

Mathematics obj=new Mathematics();

obj.Sum(); /\* no return type\*/

}

}

14. calling method with return and no parameter

package examplePack;

public class Mathematicaly

{

int Sum()

{

int a=10;

int b=20;

int c=a+b;

return c;

}

public static void main(String[] args)

{

Mathematicaly obj=new Mathematicaly();

int r=obj.Sum(); /\* No parameter with return type \*/

System.out.println("Addition values : "+r);

}

}

15. with parameter and no return type:

package examplePack;

public class Mathematicaly

{

void Sum(int x,int y)

{

int a=x;

int b=y;

int c=a+b;

System.out.println("Addition values : "+c);

}

public static void main(String[] args)

{

Mathematicaly obj=new Mathematicaly();

obj.Sum(100,300); /\*parameter with no return\*/

}

}

16. calling method with return and parameter

package examplePack;

public class Mathematicaly

{

int Sum(int x,int y)

int a=x;

int b=y;

int c=a+b;

return c;

}

public static void main(String[] args)

{

// Declare the class object

Mathematicaly obj=new Mathematicaly();

int r=obj.Sum(100,300); /\*parameter with return type\*/

System.out.println("Addition value : "+r);

}

}

calling static method

using static property: it will maintain

**Ans:** public static int empid;