

## 26912-WGHCBANDARA

### JAVA TUTORIAL

#### 01) Tutorial 01

```
package com.mycompany.javapractical01;
```

```
public class JavaPractical01  
{  
    public static void main(String[] args)  
    {  
        System.out.println("Hello World!");  
    }  
}
```

Pom.xml

```
<?xml version="1.0" encoding="UTF-8"?>  
  
<project xmlns="http://maven.apache.org/POM/4.0.0"  
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
  
    <modelVersion>4.0.0</modelVersion>  
  
    <groupId>com.mycompany</groupId>  
  
    <artifactId>JavaPractical01</artifactId>  
  
    <version>1.0-SNAPSHOT</version>  
  
    <packaging>jar</packaging>  
  
    <properties>  
  
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>  
  
    <maven.compiler.source>20</maven.compiler.source>  
  
    <maven.compiler.target>20</maven.compiler.target>  
  
    <exec.mainClass>com.mycompany.javapractical01.JavaPractical01</exec.mainClass>
```

</properties>

</project>

## 02)Tutorial 02

Item.java

```
package com.mycompany.javapractical02;
```

```
public class Item
{
    private int location;
    private String description;
    public Item(int a, String b)
    {
        location = a;
        description = b;
    }
    public void setLocation(int c)
    {
        location = c;
    }
    public void setDescription(String d)
    {
        description = d;
    }
    public int getLocation()
    {
        return location;
    }
}
```

```
public String getDescription()
{
return description;
}
}
```

Javapractical02.java

```
package com.mycompany.javapractical02;
```

```
public class JavaPractical02
{
public static void main(String[] args)
{
Monster m1 = new Monster(10, "Hello");
System.out.println("Location "+m1.getLocation());
System.out.println("Description "+m1.getDescription());
}
}
```

Monster.java

```
/*
* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
* Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
*/
package com.mycompany.javapractical02;
```

```

/**
 *
 * @author sithu
 */
public class Monster extends Item
{
    public Monster(int a, String b)
    {
        super(a,b);
    }
}

```

#### PART 02

- Which of these keywords is used to refer to member of base class from a sub class?  
a) upper    **b) super**    c) this    d) None of the mentioned
- The modifier which specifies that the member can only be accessed in its own class is  
a) public    **b) private**    c) protected    d) none
- Which of these is a mechanism for naming and visibility control of a class and its content?  
a) Object    **b) Packages**  
c) Interfaces    d) None of the Mentioned.
- Which of the following is correct way of importing an entire package 'pkg'?  
a) import pkg.    b) import pkg.\*  
**c) import pkg.\***    d) Import pkg.\*
- Which of these method of class String is used to extract a single character from a String object?  
a) CHARAT()    b) charatt()  
**c) charAt()**    d) CharAt()
- Which of these method of class String is used to obtain length of String object?  
a) get()    b) Sizeoff()  
c) lengthof()    **d) length()**

#### PART 03: Fill in the blanks using appropriate term.

- Real-world objects contain **state** and **behaviour**.
- A software object's state is stored in **fields**.
- A software object's behavior is exposed through **methods**.
- Hiding internal data from the outside world, and accessing it only through publicly exposed methods is known as data **encapsulation**.
- A blueprint for a software object is called a **class**.
- Common behavior can be defined in a **superclass** and inherited into a **subclass** using the **extends** keyword.
- A collection of methods with no implementation is called an **interface**.

- 
- A namespace that organizes classes and interfaces by functionality is called a **package**.
  - The term API stands for **application programming interface**.

### 03)Tutorial 03

Encapsulatedcalss.java

```
package com.mycompany.javapractical03;
```

```
public class EncapsulatedClass
{
    private String name;
    private int age;
    private double salary;
    public EncapsulatedClass(String i, int j,double k)
    {
        name = i;
        age = j;
        salary = k;
    }
    public void setName(String a)
    {
        name = a;
    }
    public void setAge(int b)
    {
        age = b;
    }
    public void setSalary(double c)
    {
        salary = c;
    }
}
```

```
}  
  
public String getName()  
{  
    return name;  
}  
  
public int getAge()  
{  
    return age;  
}  
  
public double getSalary()  
{  
    return salary;  
}  
}
```

## Practical

```
package com.mycompany.javapractical03;
```

```
public class JavaPractical03  
{  
    public static void main(String[] args)  
    {  
        EncapsulatedClass e1 = new EncapsulatedClass("Sithu", 20, 350000.0);  
        System.out.println("Name = "+e1.getName());  
        System.out.println("Age = "+e1.getAge());  
        System.out.println("Salary = "+e1.getSalary());  
        e1.setName("Hansa");  
    }  
}
```

```
e1.setAge(10);  
e1.setSalary(25000.50);  
System.out.println("Name = "+e1.getName());  
System.out.println("Age = "+e1.getAge());  
System.out.println("Salary = "+e1.getSalary());  
}  
}
```

#### **04)Tutorial 04**

Employee.java

```
package com.mycompany.javapractical04;
```

```
public class Employee  
{  
    private int empID;  
    private String empName;  
    private String empDesignation;  
    public void setEmpID(int a)  
    {  
        empID = a;  
    }  
    public int getEmpID()  
    {  
        return empID;  
    }  
    public void setEmpName(String b)  
    {
```

```

empName = b;
}
public String getEmpName()
{
return empName;
}
public void setEmpDesignation(String c)
{
empDesignation = c;
}
public String getEmpDesignation()
return empDesignation;
    }
}

```

Javapractical04.java

```

package com.mycompany.javapractical04;

```

```

public class JavaPractical04
{
public static void main(String[] args)
{
Employee e1 = new Employee();
e1.setEmpID(100);
e1.setEmpName("Bodgan");
e1.setEmpDesignation("Yes");
System.out.println("Emp ID = "+e1.getEmpID());
System.out.println("Emp Name = "+e1.getEmpName());
}
}

```



```

System.out.println("Emp Designation = "+e1.getEmpDesignation());

Employee e2 = new Employee();

e2.setEmpID(101);

e2.setEmpName("Bird");

e2.setEmpDesignation("No");

System.out.println("Emp ID = "+e2.getEmpID());

System.out.println("Emp Name = "+e2.getEmpName());

System.out.println("Emp Designation = "+e2.getEmpDesignation());

}

}

```

Prom.xml

```

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.mycompany</groupId>

<artifactId>JavaPractical04</artifactId>

<version>1.0-SNAPSHOT</version>

<packaging>jar</packaging>

<properties>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

<maven.compiler.source>20</maven.compiler.source>

<maven.compiler.target>20</maven.compiler.target>

<exec.mainClass>com.mycompany.javapractical04.JavaPractical04</exec.mainClass>

</properties>

</project>

```

