

|   |                        |
|---|------------------------|
| Agenda/Outcome:-  | Basics (Non OO)        |
|   | OOP Foundation         |
|   | Classes & Objects      |
|   | Inheritance            |
|   | Interfaces             |
|   | Packages               |
| Active Learning/Student-Centric   | Exception Handling     |
| - Flipped Learning  |                        |
| - PBL/PjBL/CBL  | Multithreading         |
| - Role Based  | Strings, Arrays        |
| - Gamification/Interactive Quiz   | Files, I/O             |
| Data Abstraction - Encapsulation + Data Hiding  | Collections            |
| Inheritance   |                        |
| Polymorphism  |                        |
| Object Relationships  |                        |
| - Inheritance : is-a  |                        |
| - Composition : part-of/contained   |                        |
| - Aggregation : has-a   |                        |
| - Association : using/related   |                        |
| <a href="https://github.com/caia-techblr/java-pbl-sample">https://github.com/caia-techblr/java-pbl-sample</a> |                        |
| Java Memory Model   | Box b1;                |
| - Stack Frame   | b1 = new Box(10,12,5); |
| - Heap  |                        |
| - Code  | int arr[];             |
|   | arr = new int[10];     |
| Where static vars are stored  |                        |
| Hands-on  |                        |
| - Array Visualization   |                        |
| - Object & Reference Visualization  |                        |
| - GC  |                        |
| Parameter passing rules   |                        |
| - Primitive type variables : passed by vlaues   |                        |
| - Objects : passed by reference   |                        |
| - References : passed by value  |                        |
| - Arrays : passed by reference  |                        |

## Polymorphic examples

- : IShape, Polygon, Triangle, Rectangle, Circle
- : IAccount, SavingsAccount, CreditAccount
  - + credit
  - + debit
- : InterestCalculator, SimpleInterest, CompoundInterest
- : ICustomer : Prepaid, Postpaid
- : Grading - types of Grading
- : Employee : Faculty, Admin, Director
- : ITrip : RoadTrip, TRainTrip, AirTrip
  - + travelTime()
  - + tariff()

|             |  |
|-------------|--|
| University  |  |
| Student     |  |
| Faculty     |  |
| Sports      |  |
| Library     |  |
| Inheritance |  |
| Association |  |
| Aggregation |  |
| Composition |  |

```
class Student extends Sports
class Student extends Library

class Student implements ISports, ILibrary

class Student
{
    Sports mSports;
    Library mLibrary;

}
```

- is-a / has-a
- Polymorphic calls
- Specialized impl vs common impl
- Optional (null/actual Object)
- Collection.sort
- Collection.min
- Collection.max

```
public static void compute(IShape ref)
{
    ref.area();
    ref.circumference();
}
```

```
compute(cir); //new Circle
compute(tr);  //new Triangle
compute(rect); //new Rectangle
```

<https://www.javatpoint.com/installation-of-log4j>

<https://www.baeldung.com/java-log4j-properties-guide>

<https://www.javatpoint.com/log4j>

SavingsAccount

+ credit

+ debit

IShape

+ area

+ circumference

CreditCardAccount

+ credit

+ debit

+ penalty

Triangle

+ area

+ circumference

+ typeOfTriable

IAccount

+ credit

+ debit

IShape tref = new Triangle()

ref.typeOfTriangle()

IAccount ref = new CreditCardAccount()

ref.penalty()

<https://github.com/caia-techblr/shapes-demo>

IShape ref;

ref = new Circle(7.0);

(or)

ref = new Rectangle(10,20);

(or)

ref = new Trianlge(3,4,5);

ref.area();

ref.circumference();

IShape ref;

int ch;

if(ch==1)

ref = new Circle(7.0);

else if(ch==2)

ref = new Rectangle(10,20);

else if(ch==3)

ref = new Trianlge(3,4,5);

ref.area();

ref.circumference();

## Apache Maven

Ref:- Apache Maven in 5 minutes (Tutorial)

<https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html>

- 1) Download maven binary file
- 2) Extract the archive (zip/tar file)
- 3) Set PATH to bin folder in extracted content
- 4) Generate template project

```
mvn archetype:generate -DgroupId=com.mycompany.app  
-DartifactId=my-app -DarchetypeArtifactId=maven-archetype-quickstart  
-DarchetypeVersion=1.5 -DinteractiveMode=false
```

```
# switch to my-app dir
```

```
mvn compile  
mvn package  
mvn test
```

is-a, has-a, part-of, related

Required classes:-

ICustomer

PrepaidCustomer

PostpaidCustomer

CustomerList

```
class Customer
{
    //id, name, bal, ph
    //type

    double getBalance()
    void credit(amount)
    {
        if type==pre
            balance += amount
        else
            balance -= amount
    }
    void makeCall(duration)
    {
        //check type..some calc
    }
}
```

Collections (java.util.\*)

ArrayList

LinkedList

Stack, Queue

HashMap, TreeMap

HashSet, TreeSet

```
IShape ref = new IShape();
```

```
IShape ref = new IShape() {
```

```
    double area() { }
```

```
    double circumference() { }
```

```
};
```