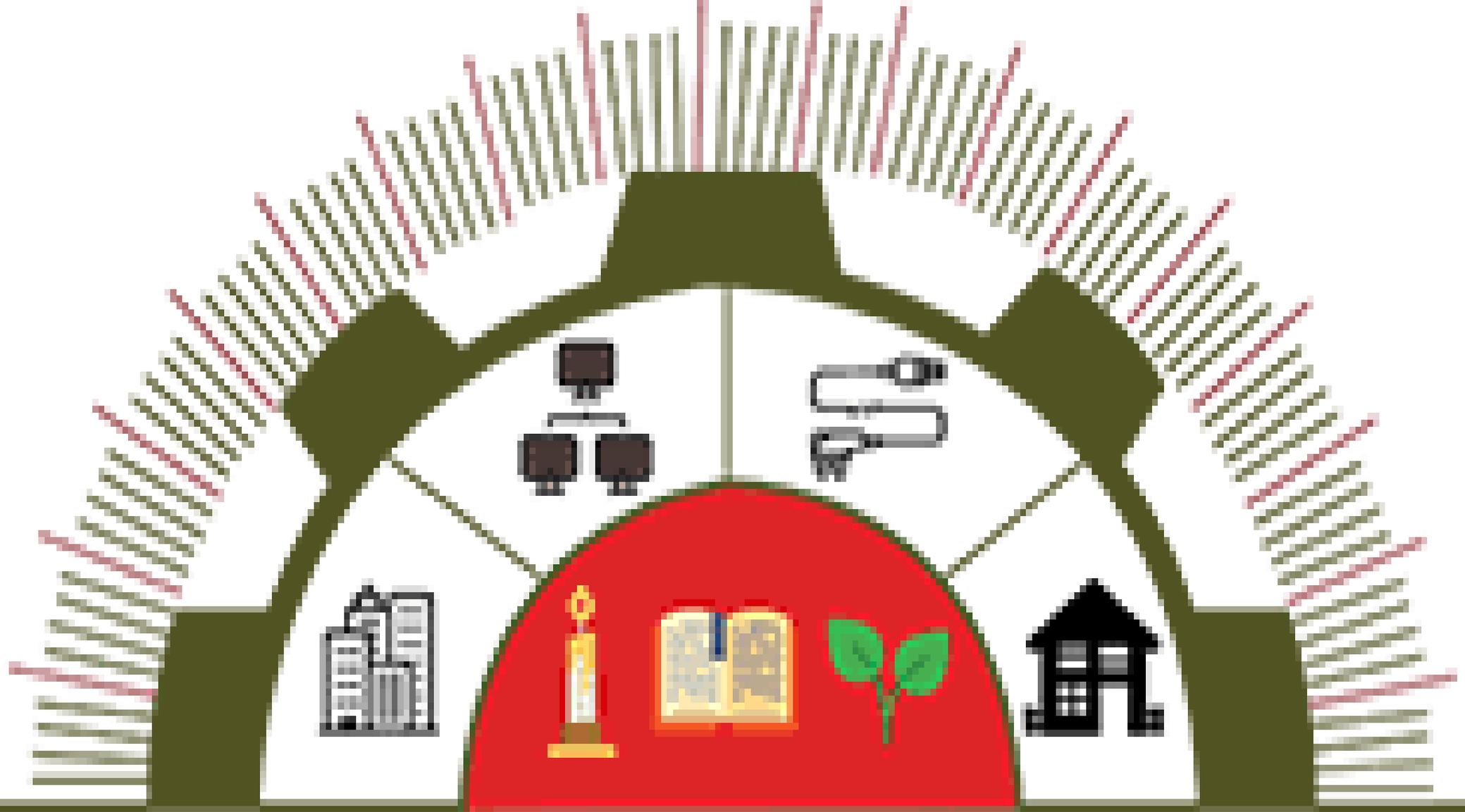


Wellcome To Our
Presentation



Habiganj Polytechnic Institute

Computer Technology

Semester: 5th

Shift: First

Group: B



Name: Md Ariful Islam

Roll: 936368

Reg: 863731

Programming In Java

Chapter Name: Inheritance and Polymorphisum

Chapter No. 5

Topics: 5.1(সুপার ক্লাস ও সাব ক্লাস)

- **ইনহেরিটেন্স** : যে প্রক্রিয়ার পূর্বে ডিফাইনকৃত এক বা একাধিক ক্লাস হতে গুনাগুন ও বিশিষ্ট ইনহেরিট করে নতুন কোন ক্লাস তৈরি করা হয়।তাকে ইনহেরিটেন্স বলে
- **সুপার ক্লাস**: ইনহেরিটেন্স প্রক্রিয়ার পূর্বে ডিফাইন কৃত যে এক বা একাধিক ক্লাস হতে বিশিষ্ট ইনহেরিট করা হয়। সে সকল ক্লাস কে সুপার ক্লাস বলে।
- **সাব ক্লাস**: ইনহেরিটেন্স প্রক্রিয়ার পূর্বে ডিফাইন কৃত এক বা একাধিক ক্লাস হতে বৈশিষ্ট্য ইনহেরিট করে নতুন যে ক্লাস তৈরি করা হয়। সেই ক্লাস কে সাব ক্লাস বলে।

Topics: 5.2(বিভিন্ন প্রকার ইনহেরিটেন্স)

- সিঙ্গেল ইনহেরিটেন্স
- মাল্টিলেভেল ইনহেরিটেন্স
- হায়ারার্কি ইনহেরিটেন্স
- সিঙ্গেল ইনহেরিটেন্স: যখন কোন সাপ প্লাস কেবল একটি ব্যাস কাছ থেকে এক বা একাধিক বিশিষ্ট ইনহেরিট করে তখন সেই প্রক্রিয়াকে সিঙ্গেল ইনহেরিটেন্স বলা হয়।
- সিঙ্গেল ইনহেরিটেন্স এর সিনট্যাক্স:

Class A

{

: // Class body

}

Class B extends A

{

M // class body

}

মাল্টিলেভেল ইনহেরিটেন্স: যখন কোন সাব ক্লাস অপর কোন সুপার ক্লাসের সাব ক্লাস থেকে একাধিক বা একাধিক বিশিষ্ট ইনহেরিট করে তখন সেই প্রক্রিয়াকে মাল্টিলেভেল ইনহেরিটেন্স বলা হয়।

মাল্টিলেভেল ইনহেরিটেন্স এর সিনট্যাক্স:

```
Class A
{
:
: //Class body
}
Class B extends A
{
:
: //Class body B
}
:
: //Class body
}
```

হায়ারার্কিক্যাল ইনহেরিটেন্স: যখন একটি মাত্র সুপার ক্লাস হতে দুই বা ততোধিক ক্লাস গঠন করা হয় তখন সেই প্রক্রিয়াকে হায়ারার্কিক্যাল ইনহেরিটেন্স বলা হয়

সিনটেক্স

```
Class A
{
//... ..
}
Class B extends A
{
//.....
}
Class C extends A
{
//.....
}
```

```
Class D extends C
{
//.....
}
Class E extends C
{
//.....
}
Class Nextends A
{
//.....
}
```

Topics: 5.3 (জাভা ভাষায় ওভাররিডেন মেথড)

ওভার রিডেনের মেথডের বৈশিষ্ট্য:

- (i).কোন একটি methodএর নাম এবং argument এর সংখ্যা,ধরন ও ক্রম যদি ছবছ parent বা superclass এর corresponding method এর মতো হয় তবে তা হলো overridden method
- (ii).প্রতিটি parent class এর method কে এর subclass এ একবার মাত্রই override করা যাবে অর্থাৎ কোন একটি Class এ signature এর দুটি method থাকবে না।
- (iii).overridden method এর return type অবশ্যই অনুরূপ হতে হবে যাকে সে override করছে। (iv).overridden method এর access modifier কখনো more private হতে পারে না। অর্থাৎ superclass এর কোনো method যদি public হয় তবে subclass এ কে কখনো protected অথবা private বা friendly করা যাবে না।

Topics: 5.6 (ফাইনাল কিওয়ার্ড)

সংজ্ঞা:

:ফাইনাল কিওয়ার্ড অনেকটা কম্পট্যান্ট ভেরিয়েবলের ন্যায় কাজ করে। অর্থাৎ final ভেরিয়েবল একবার কোন ডাটা সংরক্ষণ করা হলে এটি আর কখনো পরিবর্তন করা যায় না।

ফাইনাল মডিফাই এর ক্ষেত্রে বিবেচ্য বিষয়গুলো:

- (i) ফাইনাল অবজেক্টের রেফারেন্স ভেরিয়েবল পরিবর্তন করা যাবে না।
- (ii) ফাইনাল অবজেক্ট এর ভিতর অবস্থিত তার পরিবর্তন করা যাবে।
- (iii) ফাইনাল মেথডকে কখনো ওভাররাইট করা যাবে না।

1 থেকে 100 পর্যন্ত সংখ্যার যোগফল নির্ণয়ের প্রোগ্রাম

```
public class Calculation
{
    public static void main(String[] args)
    {
        int sum=0;
        int i;
        for(i=1; i<=100; i++)
        {
            System.out.println(i);
            sum=sum+1;
        }
        System.out.println(sum);
    }
}
```

জোড় বিজোড় সংখ্যা নির্ণয়ের প্রোগ্রাম:

- /*
 - * To change this license header, choose License Headers in Project Properties.
 - * To change this template file, choose Tools | Templates
 - * and open the template in the editor.
 - */
 - package hello.world;

 - /**
 - *
 -
 - }
- * @author Mohammad Ariful Islam
 - */
 - public class HelloWorld {

 - /**
 - * @param args the command line arguments
 - */
 - public static void main(String[] args) {
 - System.out.println("I Love You");
 - // TODO code application logic here
 - }

এরিথমেটিক অপারেটর ব্যবহার করে একটিজাভা প্রোগ্রাম:

```
➤ Class program {  
➤     Static void Main (string []args){  
➤         int a=30;  
➤         int b=10;  
➤         int C=;  
➤     C=a+b;  
➤     System.out.println("addition="+C);  
➤     C=a-b;  
➤     System.out.println("Subtraction="-C);  
➤     C=a*b;  
➤     System.out.println("Multication="+C);  
➤     C=a/b;  
➤     System.out.println(Division="+C);  
➤     C=a%b;  
➤     System.out.println("Modulus="+C  
);  
➤     C=a++;  
➤     System.out.println("Increment="+  
C);  
➤     C=a--  
➤     System.out.println("Decrement=""  
+C);  
➤     }  
➤ }
```

The Presentation Is End

Thanks For Watching