

Quick & Easy Way

Your Professional Trainer

অবজেক্ট ওরিয়েন্টেড পিএইচপি

From Novice To Professional

OOE



সহজ ভাষায় সবার জন্য

Best web programming language

Bookbd Series...

বর্তমান সময়ে সবচেয়ে জনপ্রিয় ল্যাঙ্গুয়েজ হচ্ছে অবজেক্ট ওরিয়েন্টেড প্রোগ্রামিং ।
তাই ওয়েব বেইজ অ্যাপ্লিকেশনে বহুল ব্যবহৃত ল্যাঙ্গুয়েজ হচ্ছে পিএইচপি ।

বুকবিডি রচিত ও সম্পাদিত

ডাউনলোড
www.bookbd.info

অবজেক্ট অরিয়েন্টেড পিএইচপি

রচনা ও সম্পাদনা : বুকবিডি

www.bookbd.info

প্রিয়,

পাঠক আমরা চেষ্টা করেছি বাংলা ভাষায় অবজেক্ট অরিয়েন্টেড পিএইচপি একটি বই লিখতে তারই ধারাবাহিকতায় এই ই-বুকটি লিখেছি। তাই আপনাদের কোন মতামত থাকলে আমাদের পাঠাবেন (info@bookbd.info) বইটিতে আমাদের আনিচ্ছাকৃত ভাবে যদি কোন ভুল থাকে তাহলে ক্ষমা সুন্দর দৃষ্টিতে দেখার জন্য অনুরোধ করছি।



প্রফেশনাল দক্ষতার জন্য
বুকবিডি সিরিজের বই সমূহ...

Bookbd Series...

ফ্রীল্যান্সিং এবং আউটসোর্সিং
তথ্যপ্রযুক্তি বিষয়ক
ই-বুক ডাউনলোড
www.bookbd.info

f /Ebookbd

উৎসর্গ : বাংলাদেশের সকল মুক্তিযোদ্ধাদের

www.bookbd.info

এক নজরে সূচিপত্র

১: অবজেক্ট অরিয়েন্টেড প্রোগ্রামিং.....	৫
২: অবজেক্ট অরিয়েন্টেড ল্যাঙ্গুয়েজের সুবিধা.....	৫
৩: বেসিক OOP.....	৬
৪: ক্লাস কি.....	৮
৫: অবজেক্ট কি.....	১০
৬: অবজেক্ট তৈরি.....	১১
৭: ক্লাস মেথড.....	১৩
৮: মেথড তৈরি (Method Declaration).....	১৪
৯: মেথড রিটার্ন এবং প্যারামিটার (Method Return and Parameter).....	১৭
১০: কনস্ট্রাক্টর (Constructor).....	২২
১১: কনস্ট্রাক্টর এ ভ্যালু পাঠানো.....	২৪
১২: ডেস্ট্রাক্টর (Destructor).....	২৬
১৩: মডিফায়ার (Modifiers).....	২৯
১৪: this কী ওয়াড.....	৩৫
১৫: ইনহেরিটেন্স (Inheritance).....	৩৬
১৬: ইনহেরিটেন্স এর সুবিধা.....	৩৭
১৭: ইনহেরিটেন্স প্রোগ্রাম.....	৩৭
১৮: পলিমরফিজম (Polymorphism).....	৪১
১৯: পলিমরফিজম প্রোগ্রাম সিনট্যাক্স.....	৪২
২০: এনক্যাপসুলেশন (Encapsulation).....	৪৫
২১: প্রোগ্রামিং এ এনক্যাপসুলেশন (Encapsulation).....	৪৫
২২: এনক্যাপসুলেশন প্রোগ্রামিং.....	৪৫
২৩: অ্যাবসট্রাক্ট (Abstract).....	৪৭
২৪: প্রোগ্রামিং অ্যাবসট্রাক্ট.....	৪৭
২৫: অ্যাবসট্রাক্ট ক্লাস সিনট্যাক্স.....	৪৭
২৬: অ্যাবসট্রাক্ট প্রোগ্রামিং.....	৪৮
২৭: ইন্টারফেস (Interface).....	৫০
২৮: ইন্টারফেস ক্লাসের সিনট্যাক্স.....	৫১
২৯: ইন্টারফেস ক্লাস প্রোগ্রাম.....	

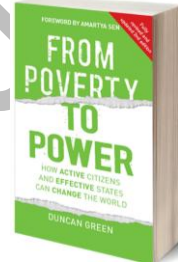
অবজেক্ট অরিয়েন্টেড প্রোগ্রামিং পিএইচপি

১: অবজেক্ট অরিয়েন্টেড প্রোগ্রামিং:

আমাদের চারদিকে আমরা কি দেখতে পাই। আমরা দেখছি যেমন গাড়ি, মানুষ, বিল্ডিং, কলম, পেন, টেবিল ইত্যাদি। অর্থাৎ আমাদের বাস্তব জীবনে আমরা চতুর্দিকে যা দেখি প্রোগ্রামিং এ তাকে অবজেক্ট বলে। আর অবজেক্ট অরিয়েন্টেড প্রোগ্রামিং ল্যাঙ্গুয়েজ হচ্ছে এমন একটি ল্যাঙ্গুয়েজ যেটি অবজেক্টকে উপস্থাপন (represent) করে। প্রোগ্রামিং এর ভাষায় অবজেক্ট অরিয়েন্টেড প্রোগ্রামিং অবজেক্ট (object) কে উপস্থাপন (represent) করে, যেটির ডেটা ফিল্ড এবং অ্যাক্টিভিউট থাকবে।



চিত্র ১: গাড়ি (অবজেক্ট)



চিত্র ২: বই (অবজেক্ট)

উপরের ছবিতে গাড়ি এবং বই দুটোই হচ্ছে অবজেক্ট।

২: অবজেক্ট অরিয়েন্টেড ল্যাঙ্গুয়েজের সুবিধা

- এটি সহজ হওয়ার কারণে সফটওয়্যার এর অবজেক্ট মডেল এর সাথে বাস্তব জীবনের অবজেক্ট এর সাথে মিল আছে।
- Object Oriented Programming (OOP) এর কোডগুলো পুনরায় ব্যবহার করা যায়। অর্থাৎ একই অবজেক্ট বিভিন্ন প্রোগ্রাম এ ব্যবহার করা যায়।
- অবজেক্ট গুলোকে খুব সহজে মেনটেইন করা যায়।
- OOP কে মডিফাই করা অনেক সহজ।
- অনেক বড় সমস্যাকে OOP এর মাধ্যমে খুব সহজে সমাধান করা যায়।
- খুব সহজে একটি কোডের/ প্রোগ্রাম এর সাথে নতুন কোন ফিচার সংযোগ করা যায়।

- প্রোথামের ডেটা এবং ফাংশনকে আবাসিত হস্তক্ষেপ থেকে সহজে রক্ষা করা যায় এবং সীমাবদ্ধতাও দিয়ে দেওয়া যায়।
- অবজেক্ট অরিয়েন্টেড ল্যাপুয়েজ ব্যবহার করলে প্রোথামের জটিলতা অনেক কমে যায়।

৩: বেসিক OOP:

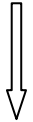
এখন আমরা কম্পিউটারের C ড্রাইভে গিয়ে htdocs এর ভিতরে গিয়ে একটি ফাইল ওপেন (open) করবো এবং ফাইলটির নাম হচ্ছে p1.php। এবার ফাইলটিতে নিম্নের কোডগুলো লিখুন:

```
class bookbd
{
}
```

প্রোথাম সিনট্যাক্স ১ : অবজেক্ট অরিয়েন্টেড ক্লাস

<?php ← php ট্যাগ শুরু করলাম

কী-ওয়ার্ড (এটি লিখতে হয়)



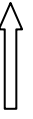
class bd ← ক্লাস এর নাম bd

{ ← দ্বিতীয় ব্র্যাকেট দিয়ে শুরু করতে হয়।

ভেরিয়েবল নেইম



public \$city="dhaka"; ← ভেরিয়েবল এর মান dhaka



মডিফায়ার (কী-ওয়ার্ড)

} ← দ্বিতীয় ব্রাকেট দিয়ে শেষ (অর্থাৎ ক্লাসটি শেষ হল)

ইকুয়াল সাইন

new এটি হচ্ছে কী-ওয়ার্ড
\$x=new bd; ← ক্লাস নেইম

এখানে ভেরিয়েবল x হচ্ছে একটি অবজেক্ট

অবজেক্ট x
echo \$x->city; ← bd ক্লাস এর ভেরিয়েবল

নেইম স্পেস/অ্যারো অপারেটর
আউটপুট দেখানোর জন্য

?> ← পিএইচপি ট্যাগ শেষ করা হল।

প্রোগ্রাম সিনট্যাক্স ২

```
<?php  
class bd  
{  
    public $city="dhaka";  
}  
  
$x = new bd;
```



```
echo $x->city;
```

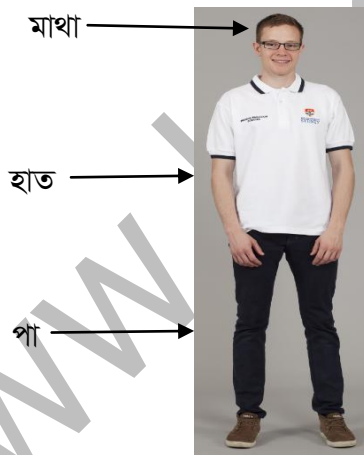
```
?>
```

প্রোগ্রাম p1.php: বেসিক oo পিএইচপি প্রোগ্রাম উপরের প্রোগ্রামটি (p1) রান করলে আউটপুট dhaka দেখাবে

আউটপুট: dhaka

৪: ক্লাস কি:

ক্লাস হচ্ছে এমন একটি জিনিস যেখানে অবজেক্টকে ডিফাইন করা হয়। অর্থাৎ অবজেক্ট সম্পর্কে সব কিছু ক্লাসে বলা থাকে। যেমন আমি যদি বলি বই একটি অবজেক্ট তাহলে বইটি সম্পর্কে সব কিছু ক্লাসে বলা থাকবে। বইটি কালার কেমন, পেইজ সংখ্যা কত, লেখক কে, মূল্য কত ইত্যাদি ক্লাসে বলা থাকবে। ক্লাস অবজেক্টটিকে রিপ্রেজেন্ট (represent) করে। অর্থাৎ অবজেক্টের এর সকল প্রোপার্টিজ এবং অ্যাট্রিবিউট (Attribute) গুলো ক্লাসে বলা থাকে। চলুন আরেকটি উদাহরণ দেখি। যেমন কার (car) হচ্ছে একটি অবজেক্ট, এখন কার অবজেক্ট এর জন্য ক্লাসটিতে বলা আছে এর কালার কি, কার এর কয়টি দরজা ইত্যাদি।



চিত্র ৩: একটি মানুষের বর্ণনা

উপরের ছবিতে মানুষটির যে দুটো হাত, লম্বা ৫.৬ ইঞ্চি, চোখ, ইত্যাদি অর্থাৎ মানুষটি সম্পর্কে যা কিছু বলবো সেটাই হল ক্লাস। সুতরাং ক্লাসে একটি অবজেক্ট সম্পর্কে বিস্তারিত বলা থাকে।

class cow → এটি হচ্ছে একটি cow ক্লাস

```
{
```

এটি হচ্ছে ক্লাস এর বডি অর্থাৎ এখানে অবজেক্ট সম্পর্কে সব কিছু এখানে বলা থাকবে।

```
}
```

নিম্নে ক্লাস এর একটি উদাহরণ দেওয়া হল

```
<?php
```

```
class opc
```

```
{
```

```
    public $name="Kamal Ahmed";
```

↑
এটি হচ্ছে ভেরিয়েবল

↑
ভেল্যু

```
}
```

ক্লাস নিজে নিজে কাজ করে না যদি না তাকে কল (call) করা না হয়। প্রোগ্রাম রান করবে কিন্তু ক্লাস এর ভিতর যে মেথড এবং প্রোপার্টিজ আছে সেগুলো কাজ করবে না। তাই এখন দেখবো কিভাবে ক্লাস কে রান (execute) করতে হয়।

ক্লাস কে রান করতে হলে প্রথমে ক্লাসের অবজেক্ট তৈরি (create) করতে হবে।

```
$ob=new opc;
```

অর্থাৎ প্রথমে যে কোন একটি ভেরিয়েবল নিতে হবে যেমন এখানে \$ob নিয়েছি। আর \$ob ই হচ্ছে opc ক্লাস এর অবজেক্ট। তারপর ইকুয়াল (=) সাইন লিখে নিয়ম অনুযায়ী new কী-ওয়ার্ডটি লিখতে হবে এর পর স্পেস (space) দিয়ে ক্লাস এর নাম লিখে সেমিকোলন ব্যবহার করতে হবে।

✓ মনে রাখবেন সাধারণত ক্লাস এবং অবজেক্ট এর নাম যে কোন কিছু হতে পারে।

চলুন এখন দেখি ঐ ob অবজেক্টটি কি কাজ করে।

প্রথমে আমরা ob অবজেক্ট দিয়ে opc ক্লাস এর ভেরিয়েবল \$name এর মান (value) দেখতে চাচ্ছি।

```
echo $ob->name;
```

এখন আমরা আউটপুট দেখবো Kamal Ahmed

অবশ্যই মনে রাখবেন এই বইটি পড়ার পূর্বে আমাদের লেখা "পিএইচপি অ্যান্ড মাইএসকিউল" বইটি অবশ্যই পড়বেন। অন্যথায় এই বইটি বুঝতে আপনার সমস্যা হতে পারে।

```
<?php
include('p1.php');

class MyClass
{
    public $pop = "I'm a class property!"; // pop variable in the class
}

$obj = new MyClass; // the way to create object

echo $obj->pop; // Output the property (want to see the output of class)

?>
```

প্রোগ্রাম নং p2.php: ভেরিয়েবল এবং ক্লাস
আউটপুট:

৫: অবজেক্ট কি:

- বাস্তব (real world) জীবনের সব কিছু অবজেক্ট। আমরা যদি আমাদের চারদিকে তাকাই তাহলে আমরা অনেক অবজেক্ট দেখতে পাবো। যেমন আমাদের চারপাশে আমরা দেখতে পাই পেন, টেবিল, মানুষ, বালক, বালিকা, কম্পিউটার, বুক, বিলবোর্ড, ব্রিজ, ফ্যান, বিল্ডিং ইত্যাদি।
- বাস্তব জীবনে প্রত্যেকটি অবজেক্ট এর দুটো জিনিস থাকে একটি হচ্ছে প্রোপার্টিজ (properties) অন্যটি হচ্ছে বিহেবিয়ারস (behaviors)
- যেমন শার্ট হচ্ছে একটি অবজেক্ট, এর প্রোপার্টি (property) হচ্ছে শার্টটির রং, সাইজ এবং কোম্পানি নেইম ইত্যাদি। অন্যদিকে (বিহেবিয়ারস) behaviors হচ্ছে শার্টটি গরম এবং শীতকালে আরামদায়ক, শার্টটি ইন করে এবং ইন ছাড়া পড়া যায়। অন্য আরেকটি উদাহরণ দেই যেমন একটি কার (car) হচ্ছে অবজেক্ট, এখন দেখি এর property কি।

কার অবজেক্টটির property হচ্ছে কারটির কালার এবং কারটির ব্রাড নেইম ইত্যাদি।
এর এটির behaviors হচ্ছে কারটি সামনে এবং পিছনে নেওয়া যায়।

- এখন দেখবো অবজেক্ট আমরা কেন প্রোগ্রামিং এ ব্যবহার করি। আমরা যখন কোন ক্লাস লেখি ঐ ক্লাসে অনেক ভেরিয়েবল, ফাংশন ইত্যাদি থাকে আর সেগুলো নিজে নিজে কাজ করে না যতক্ষণ না পর্যন্ত অবজেক্ট দ্বারা ঐ গুলিকে আমরা কল (ডাকব) না করি। ক্লাসের ভেরিয়েবলকে ঐ ক্লাসের অবজেক্ট দ্বারা কাজ করতে হবে। অবজেক্ট তৈরি করতে হয় কোন একটি ক্লাসের জন্য। যেমন বই যদি একটি ক্লাস হয় তাহলে একটি অবজেক্ট বলা যেতে পারে ইতিহাস। এখানে ইতিহাস হচ্ছে বুক ক্লাসের অবজেক্ট। প্রিয় পাঠক এখানে আমি লেখাগুলোকে স্বাভাবিক ভাষায় লেখার চেষ্টা করেছি সে জন্য হয়তো অনেক ভালো মানের বাংলা ব্যাকরণ সংযুক্ত লেখা হয় নাই, সেজন্য দুঃখিত।

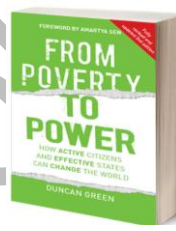
নিম্নে কতগুলো অবজেক্ট এর ছবি দেওয়া হল।



চিত্র ৪: পেন



চিত্র ৫: গ্লাস



চিত্র ৬: বই



চিত্র ৭: চেয়ার

৬: অবজেক্ট তৈরি:

অবজেক্ট তৈরি (create) করতে হলে কোন না কোন একটি ক্লাস থাকতে হবে। এখন আমি আপনাদেরকে ম্যান (man) ক্লাস এর একটি অবজেক্ট তৈরি করে দেখাবো। অবজেক্ট তৈরি (create) করতে হলে প্রথমে ডলার সাইন (\$) দিয়ে যে কোন একটি নাম দিতে হবে যেটি হচ্ছে অবজেক্ট নেইম তারপর ইকুয়াল (=) সাইন ব্যবহার করতে হবে এরপর নিউ (new) নামক কী-ওয়ার্ডটি ব্যবহার করতে হবে তারপর ক্লাসের নামটি (যে ক্লাসের জন্য অবজেক্টটি) লিখতে হবে এরপর দুটো প্রথম ব্রাকেট () দিয়ে তারপর সেমিকোলন (;) দিয়ে শেষ করতে হবে।

\$o=new man();

উপরের \$o হচ্ছে man ক্লাস এর অবজেক্ট

```
$object=new man();
```

```
$m=new man();
```

```
$Rakib=new man();
```

উপরে \$object, \$m, \$Rakib হচ্ছে man ক্লাস এর অবজেক্ট । আপনি অবজেক্ট এর নাম যে কোন কিছু দিতে পারবেন ।

```
<?php  
  
class human { // Name of class  
public $gender='Male'; // in the gender variable assign male value  
}  
$manik = new human(); // create object manik  
echo $manik->gender; // access the variable gender using human class object  
//manik  
  
?>
```

প্রোগ্রাম p4.php: ক্লাস এবং অবজেক্ট ।

আউটপুট: Male

মনে রাখবেন বাবা ছাড়া যেমন সন্তান হতে পারে না, ঠিক তেমনি ক্লাস ছাড়া অবজেক্ট হতে পারে না । অনেক সময় ছেলের নাম রাখার সময় বাবার নামের সাথে মিল রেখে রাখে । কিন্তু প্রোগ্রামিং এ অবজেক্ট তৈরি করার সময় অবশ্যই ক্লাস এর নাম তখন থাকে । সন্তান এর নাম যেমন যে কোন কিছু হয় অবজেক্ট এর নামও আপনি যে কোন কিছু দিতে পারেন ।

ই-কমার্স সম্পর্কে বিস্তারিত জানতে আমাদের লেখা "ই-কমার্স" বইটি পড়তে পারেন । বইটিতে ই-কমার্স সম্পর্কে বিস্তারিত বলা আছে ।

```
<?php  
  
class Kool {  
    public $mvar = 'aMemberVar Member Variable';  
    public $aFuncName = 'Function of member';  
  
    function mfunc() {  
        print 'Mfunc is the inside of kool class';  
    }  
}
```

```
$kk = new kool;  
echo $kk->mfunc();  
?>
```

প্রোগ্রাম p5.php: ক্লাস এবং অবজেক্ট।

আউটপুট: Mfunc is the inside of kool class

৭: ক্লাস মেথড:

মেথড হচ্ছে এক ধরনের ফাংশন। নির্দিষ্ট কিছু কাজ মেথড করে থাকে। অর্থাৎ মেথড সব সময় একই ধরনের কাজ করে থাকে যার জন্য একই ধরনের কোন কাজ করার জন্য যেন একই কোড বার বার লিখতে না হয়। সে জন্য আমরা মেথড ব্যবহার করতে পারি। মনে রাখবেন আমরা চেষ্টা করছি লেখাগুলোকে প্রচলিত শব্দে লিখতে সে জন্য হয়তো সঠিক শব্দ ব্যবহার হচ্ছে না সে জন্য আমরা দুঃখিত। মেথডকে সাধারণত আমরা ক্লাসের ভিতরে লিখে থাকি। আপনি চাইলে কোন তথ্য ক্লাস এর বাইরে থেকে ক্লাসের মেথড এর ভিতর পাঠাতে পারেন। তবে মনে রাখতে হবে মেথড অথবা ফাংশন নিজে নিজে রান (execute) করে না যতক্ষণ না পর্যন্ত এটিকে কল (call) করা না হয়। মেথড অবজেক্ট কোন একটি নির্দিষ্ট কাজ করতে সাহায্য করে। উদাহরণ হিসেবে বলা যায় মাউস ক্লিক অনেকটা মেথড এর মত। যেমন আমরা যখন ফোল্ডার ওপেন করি মাউস ক্লিক করি, যখন কোন ফাইল ওপেন করি মাউস ক্লিক করি, যখন কোন ব্রাউজার ওপেন করি মাউস দিয়ে ক্লিক করি অর্থাৎ এখানে সব কাজের জন্য মাউস ক্লিক একটি কমন কাজ।



চিত্র ৮: ব্লেন্ডার মেশিন



চিত্র ৯: গ্লাস

একই মেশিন ব্যবহার করে আপনি অনেক ধরনের জুস তৈরি করতে পারেন ঠিক তেমনি একই গ্লাস ব্যবহার করে অনেকে কিছু পান করতে পারেন। এমনি অনেকে একই গ্লাস ব্যবহার করতে পারেন।

অবশ্যই মনে রাখবেন এই বইটি পড়ার পূর্বে আমাদের লেখা পিএইচপি অ্যান্ড মাইএসকিউল বইটি অবশ্যই পড়বেন। অন্যথায় এই বইটি বুঝতে আপনার সমস্যা হতে পারে।

৮: মেথড তৈরি (Method Declaration):

আগেই বলেছি মেথড ডিক্লারেশন হচ্ছে অনেকটা ফাংশন এর মত। প্রথমে যে কোন একটি ক্লাস ডিক্লেয়ার করতে হবে তারপর ক্লাস এর ভিতর মেথডটি লিখতে হবে। প্রথমে ফাংশন (function) নামক কী-ওয়ার্ডটি লিখতে হবে তারপর মেথড এর জন্য একটি নাম লিখতে হবে, এরপর দুটি প্রথম ব্রাকেট () দিতে হবে, এরপর একটি সেকেন্ড ব্রাকেট ({} এরপর মেথড এর ভিতর যা লিখতে চান লিখার পর আরেকটি সেকেন্ড ({} ব্রাকেট দিয়ে শেষ করতে হবে।

```
function glass
```

```
{
```

```
}
```

প্রোগ্রাম সিনট্যাক্স ৩ : ফাংশন (মেথড)

ফাংশন কী-ওয়ার্ড

মেথড নেইম

function Room() ← ব্রাকেট

{ ← সেকেন্ড ব্রাকেট শুরু/মেথড শুরু

মেথড বডি

echo "this is class room";

}

মেথড শেষ

প্রোগ্রাম p6: মেথড ডিক্লারেশন

```
<?php
class university // class name university
{
    function Room() // method name Room //
    {
        //Start method
        echo "This is class room"; // method statement // Execute only if call this method outside of
this class
    }
        //End method

}

$obj = new university(); // after class you can use first bracket or not use

echo $obj->Room(); // Call method, method will not execute if you dont call //method

?>
```

প্রোগ্রাম p6.php:মেথড ডিক্লারেশন

আউটপুট: This is class room

```
<?php
class university // class name university
{
    function course() // method name Room //
    {
        //Start method
        echo "Software Engineering "; // method statement // Execute only if
call this method outside of this class
    }
        //End method

}

$obj = new university(); // after class you can use first bracket or not use
```

```
echo $obj->course(); // Call method, method will not execute if you dont call //method  
?>
```

প্রোগ্রাম p7.php: মেথড কলিং

আউটপুট: Software Engineering

```
<?php  
  
class university  
{  
    public $name = 'John'; // declare varibale name and value is john  
  
    function getName() // method name getName // function is keyword  
    {  
        //return $this->name;  
        echo $this->name; // when call getName method it will display John  
    }  
}  
$obj = new university();  
$obj->getName();  
?>
```

প্রোগ্রাম p8.php: মেথড এবং ভেরিয়েবল

আউটপুট: John

```
<?php  
  
class university  
{  
    public $name = 'Mohsin'; // declare varibale name and value is john  
  
    function getName() // method name getName // function is keyword  
    {  
        //return $this->name;  
        echo $this->name; // when call getName method it will display John  
    }  
}
```

```
$obj=new university();  
$obj->getName();
```

```
?>
```

প্রোগ্রাম p9.php: ক্লাস ভেরিয়েবল এবং মেথড

আউটপুট: Mohsin

৯: মেথড রিটার্ন এবং প্যারামিটার (Method Return and

আপনি যদি মেথডে কোন ভ্যালু পাঠাতে চান তাহলে কি ভাবে পাঠাবেন। সেজন্য মেথড এর ব্রাকেটের () এর ভিতর ভেরিয়েবল লিখতে হবে। আর মেথড কল করার সময় ভ্যালুটি দিয়ে দিতে হবে।

penType(\$price) ← এটি হচ্ছে প্যারামিটার

\$obj->penType(2) ← প্যারামিটার ভ্যালু

```
<?php  
class pen  
{  
  
    function PenType($price) // $price is parameter its vlaue come when call it  
    {  
        echo $price; // display the value of $price  
    }  
}  
$obj=new pen();  
$obj->PenType(2); // passing value of $price, value is 2  
?>
```

প্রোগ্রাম p10.php: মেথড প্যারামিটার

আউটপুট: 2

```
<?php  
class pen  
{
```

```
function PenType($price, $penName) // if more than one parameter write using comma
{
    echo $price; // display the value of $price
    echo $penName;
}

}
$obj = new pen();
$obj->PenType(13, "Econo"); // if want to pass more parameter value write using comma
?>
```

প্রোগ্রাম p11.php: একের অধিক প্যারামিটার

আউটপুট: 13Econo

```
<?php

class pen
{

    function PenType($price, $penName, $companyName) // if more than one parameter write
    using comma
    {
        echo $price; // display the value of $price
        echo $penName;
        echo $companyName;
    }

}

$obj = new pen();
$obj->PenType(13, "Econo", "Ocan"); // if want to pass more parameter value write using
comma

?>
```

প্রোগ্রাম p12.php:Parameter পাঠানো

আউটপুট: 13EconoOcan

- মেথড থেকে আমরা কিভাবে কোন রেজাল্ট পেতে পারি। অর্থাৎ মেথড এর কোন ভ্যালু আমরা নিতে চাই যেটা বলে মেথড রিটার্ন। অর্থাৎ মেথড কোন একটি কাজ করে তার ভ্যালুটা যদি ক্লাস অথবা মেথড এর বাইরে পাঠায় সেটাই হল মেথড রিটার্ন ভ্যালু।
- মেথড ভ্যালু পাঠানোর (Return) করার জন্য Return কী-ওয়ার্ডটি ব্যবহার করে। অর্থাৎ Return 5; এখানে 5 কে পাঠাচ্ছে যেখান থেকে কল করা হচ্ছে সেখানে।

```
<?php
```

```

class pen
{

    function PenType()
    {
        return 5; // return value 5 // use return keyword to reutrn vlaue
    }

}
$obj = new pen();
echo $obj->PenType(); // Here we will get 5 becuase call PenType from here

?>

```

প্রোগ্রাম p13.php: রিটার্ন মেথড ভ্যালু

আউটপুট: 5

```

<?php

class pen
{

    function PenType()
    {
        return 45.23; // return value 5 // use return keyword to reutrn vlaue
    }

}
$obj = new pen();
echo $obj->PenType(); // Here we will get 5 becuase call PenType from here

?>

```

প্রোগ্রাম p14.php: মেথড রিটার্ন ভ্যালু

আউটপুট: 45.23

```

<?php

class pen
{

    function PenType()
    {
        return "Econo"; // return string type value like Econo
    }

}

```

```
}
$obj = new pen();
echo $obj->PenType(); // Here we will get Econo

?>
```

প্রোগ্রাম p15.php: স্ট্রিং রিটার্ন

আউটপুট: Econo

```
<?php

class pen
{
    public $price=50;
    function PenType()
    {
        return $this->price; // return string type value like Econo
    }
}

$obj = new pen();
echo $obj->PenType(); // Here we will get Econo

?>
```

প্রোগ্রাম p16.php: মেথড রিটার্ন ভেরিয়েবল

আউটপুট: 50

বুকবিডি সম্পাদিত অন্যান্য বইসমূহ

- © বিগীনিং জুমলা © অ্যাডভান্সড জুমলা © প্রফেশনাল জুমলা © জুমলা টেম্পলেট মেকিং
- © বিগীনিং ওয়ার্ডপ্রেস © অ্যাডভান্সড ওয়ার্ডপ্রেস © প্রফেশনাল ওয়ার্ডপ্রেস © ইন্টারনেটে
আত্মকর্মসংস্থান-১ © ইন্টারনেটে আত্মকর্মসংস্থান-২ © ই-কমার্স অ্যান্ড জুমলা! ভার্সিটি
- © ই-কমার্স © ম্যাক্রোমিডিয়া ড্রিমওয়েভার © সার্চ ইঞ্জিন অপটিমাইজেশন © অ্যাডভান্সড
সার্চ ইঞ্জিন অপটিমাইজেশন © প্রফেশনাল সার্চ ইঞ্জিন অপটিমাইজেশন © ফরেক্স ট্রেডিং
- © ই-মার্কেটিং © অ্যাডভান্সড এইচটিএমএল © পিএইচপি অ্যান্ড মাই এসকিউএল
- © অ্যাডভান্সড পিএইচপি অ্যান্ড মাই এসকিউএল © অবজেক্ট অরিয়েন্টেড পি.এইচ.পি
ডেটাবেইজ মাই এসকিউএল © সি প্রোগ্রামিং © জাভা প্রোগ্রামিং © জাভাস্ক্রিপ্ট
- © অ্যাডভান্সড ফটোশপ © অ্যাডভান্সড ইলাস্ট্রেটর © প্রফেশনাল গ্রাফিক্স ডিজাইন
- © প্রফেশনাল কম্পিউটার নেটওয়ার্কিং © ওডেসক এবং আউটসোর্সিং © অ্যাফিলিয়েট মার্কেটিং

www.bookbd.info

বুকবিডি হচ্ছে বাংলাদেশী প্রফেশনাল বাংলা বই সমূহের ওয়েবসাইট। যেখান থেকে আপনি ই-বুক বিনা মূল্যে ডাউনলোড করতে পারবেন। যে বইগুলো আপনাকে আইটি আউটসোর্সিং-এ প্রফেশনাল ক্যারিয়ার গড়ে তুলতে সাহায্য করবে। এ ছাড়াও নিজে নিজে কোন প্রকার ট্রেনিং ছাড়াই যে কোন বিষয়ের উপর প্রফেশনাল দক্ষতা অর্জন করতে পারেন এই বইগুলো পড়ে। আর আপনাদের কোন পরামর্শ থাকলে অবশ্যই আমাদেরকে জানাবেন, এই ঠিকানায় :- info@bookbd.info

১০: কনস্ট্রাকটর (Constructor)

Constructor হচ্ছে বিশেষ ধরনের একটি ফাংশন। ফাংশনকে কল করলে এটি কাজ করে কিন্তু Constructor অটোমেটিক্যালি কল হয়। অর্থাৎ যখন আমরা কোন ক্লাসের অবজেক্ট তৈরি করি তখনই কিন্তু ঐ ক্লাসের Constructor অটো (নিজেই) রান করতে থাকে।

কনস্ট্রাকটর এবং মেথড এর মধ্যে পার্থক্য হচ্ছে অবজেক্ট তৈরির (create) সময় Constructor অটোমেটিক্যালি কল হয় আর সেই অবজেক্ট দিয়ে মেথড অথবা ফাংশনকে কল করতে হয়।

- কনস্ট্রাকটর ডিক্লেয়ার করার নিয়ম হচ্ছে প্রথমে function কী-ওয়ার্ডটি লিখতে হবে, তারপর আন্ডারস্কোর (_) ব্যবহার করবেন এরপর Construct কী-ওয়ার্ডটি লিখবেন এবং তার সাথে দুটো প্রথম ব্রাকেট () ব্যবহার করতে হবে।
নিম্নে দেখানো হচ্ছে কিভাবে Constructor লিখতে হয়

```
function_construct()  
{  
  
}
```

প্রোগ্রাম সিনট্যাক্স ৪ : কনস্ট্রাকটর

অর্থাৎ আমরা কোন কিছু লিখতে চাইলে Constructor বডিতে লিখবো।

```
<?php  
  
class human { // { this is starting bracket 1 for this class  
  
function __construct() //  
{  
  
    echo "This is Bangladesh"; // statement inside the constructor  
  
}  
  
} // { this is ending bracket 2 for this class human
```

```
$akash = new human(); // call constructor automatically when create object
// Do not need to call separately constructor
```

?>

প্রোগ্রাম p17.php:বেসিক Constructor

আউটপুট: This is Bangladesh

```
<?php
```

```
class human { // { this is starting bracket 1 for this class
```

```
function __construct() //
{
```

```
    echo "This is Bangladesh"; // statement inside the constructor
```

```
}
```

```
} // { this is ending bracket 2 for this class human
```

```
$akash = new human(); // call constructor automatically when create object
// Do not need to call separately constructor
```

?>

প্রোগ্রাম p17.php:কনস্ট্রাক্টর প্রোগ্রাম

আউটপুট: This is Bangladesh

```
<?php
```

```
class human { // { this is starting bracket 1 for this class
    public $p=0;
```

```
function __construct()
```

```
{
```

```
    $p=15;
```

```
    echo $p;
```

```
}
```

```
} // { this is ending bracket 2 for this class human
```

```
$akash = new human(); // call constructor automatically when create object  
// Do not need to call separately constructor
```

```
?>
```

প্রোগ্রাম p18.php: কনস্ট্রাক্টর উদাহরণ

আউটপুট: 15

১১: কনস্ট্রাক্টর এ ভ্যালু পাঠানো:

Constructor কিন্তু ভ্যালু গ্রহণও করে প্যারামিটার হিসেবে। যেমনটি মেথড ভ্যালু গ্রহণ করে।
তবে Constructor এ ভ্যালু পাঠাতে হবে অবজেক্ট Create করার সময়।

```
function _Construct($Name) ← parameter ১টি
```

```
{
```

```
}
```

```
function _Construct($Name,$Age) ← parameter ২টি
```

✓ মনে রাখবেন Constructor এ একের অধিক প্যারামিটার থাকতে পারে।

```
<?php
```

```
class human { // { this is starting bracket 1 for this class
```

```
function __construct($name)
```

```
{
```

```
echo $name;
```

```

}

} // { this is ending bracket 2 for this class human

$akash = new human('Jebin');// call constructor automatically pass value jebin
// Do not need to call separately constructor

?>

```

প্রোগ্রাম p19.php:একটি ভ্যালু পাঠানো

আউটপুট: Jebin

```

<?php

class human { // { this is starting bracket 1 for this class

function __construct($name, $age) // name and age two parameter value come from when create
object
{

    echo $name;
    echo $age;

}

} // { this is ending bracket 2 for this class human

$akash = new human('Jebin', 21);// call constructor automatically pass value jebin
// passing two vlaue jebing and 21 during object creation

?>

```

প্রোগ্রাম p20.php:কনস্ট্রাক্টরে একের অধিক ভ্যালু

আউটপুট: Jebin21

১২: ডেস্ট্রাকটর (Destructor):

আমরা যখন কোন অবজেক্টকে destroy (ধ্বংস) করতে চাই তখন ডেস্ট্রাকটর ব্যবহার করে থাকি। অর্থাৎ কোন ক্লাস এর তৈরি (created) অবজেক্ট যদি আপনি শেষ করে দিতে চান অর্থাৎ ভবিষ্যৎ এ এটি আপনার আর কখনো প্রয়োজন পড়বে না তখন আপনি Destructor ব্যবহার করে অবজেক্টটিকে destroy করে দিতে পারেন। পিএইচপিতে এটিকে Garbage Collector দিয়ে করা হয়। ডেস্ট্রাকটরও এক ধরনের মেথড। Constructor এর কাজ হচ্ছে অবজেক্ট তৈরি করা আর Destructor এর কাজ হচ্ছে সেই অবজেক্টকে শেষ করে দেওয়া। অবজেক্ট যখন destroy হয় destructor তখন অটোমেটিক্যালী কল হয়।

```
<?php
class X
{
    public function __construct() // This is constructor
    {
        echo 'building an object';
    }

    public function __destruct() // destructor to destroy object
    {
        echo 'destroying the object';
    }
}

$obj = new X(); // when create object both construct and destruct call automatically
?>
```

প্রোগ্রাম p21.php: অবজেক্ট এবং destroy

আউটপুট: building an object
destroying the object

```
<?php
class Yea
{
    public function __destruct()
    {
        echo 'destroy the object';
    }
}
```

```

}

$obj = new Yea();

// prints Bangladesh
echo 'Banglade4sh ';

?>

```

প্রোগ্রাম p22.php:destroy অবজেক্ট

আউটপুট: Banglade4sh destroy the object

- ✓ মনে রাখতে হবে সকল কাজ শেষ করার পর অবজেক্ট destroy করতে হবে অর্থাৎ destruct মেথডটি লিখতে হবে ক্লাসের শেষে যেন তার আগে অন্যান্য কাজগুলো শেষ হয়ে যায়।

```

<?php
class A
{
    public function printMsg()
    {
        echo 'I still exist';
    }

    public function __destruct()
    {
        echo 'destroying the object';
    }
}

$b = new A();

$d = $c = $b;

$b = null;

$c->printMsg();
$d->printMsg();

?>

```

প্রোগ্রাম p23.php:ক্লাস শেষে destruct মেথড

আউটপুট: I still existI still existdestroying the object

বুকবিডি সম্পাদিত অন্যান্য বইসমূহ

- ⊙ বিগীনিং জুমলা ⊙ অ্যাডভান্সড জুমলা ⊙ প্রফেশনাল জুমলা ⊙ জুমলা টেম্পলেট মেকিং
- ⊙ বিগীনিং ওয়ার্ডপ্রেস ⊙ অ্যাডভান্সড ওয়ার্ডপ্রেস ⊙ প্রফেশনাল ওয়ার্ডপ্রেস ⊙ ইন্টারনেটে
- আত্মকর্মসংস্থান-১ ⊙ ইন্টারনেটে আত্মকর্মসংস্থান-২ ⊙ ই-কমার্স অ্যান্ড জুমলা! ভার্সিটি
- ⊙ ই-কমার্স ⊙ ম্যাক্রোমিডিয়া ড্রিমওয়েভার ⊙ সার্চ ইঞ্জিন অপটিমাইজেশন ⊙ অ্যাডভান্সড
- সার্চ ইঞ্জিন অপটিমাইজেশন ⊙ প্রফেশনাল সার্চ ইঞ্জিন অপটিমাইজেশন ⊙ ফরেক্স ট্রেডিং
- ⊙ ই-মার্কেটিং ⊙ অ্যাডভান্সড এইচটিএমএল ⊙ পিএইচপি অ্যান্ড মাই এসকিউএল
- ⊙ অ্যাডভান্সড পিএইচপি অ্যান্ড মাই এসকিউএল ⊙ অবজেক্ট অরিয়েন্টেড পি.এইচ.পি
- ডেটাবেইজ মাই এসকিউএল ⊙ সি প্রোগ্রামিং ⊙ জাভা প্রোগ্রামিং ⊙ জাভাস্ক্রিপ্ট
- ⊙ অ্যাডভান্সড ফটোশপ ⊙ অ্যাডভান্সড ইলাস্ট্রেটর ⊙ প্রফেশনাল গ্রাফিক্স ডিজাইন
- ⊙ প্রফেশনাল কম্পিউটার নেটওয়ার্কিং ⊙ ওডেস্ক এবং আউটসোর্সিং ⊙ অ্যাফিলিয়েট মার্কেটিং

www.bookbd.info

বুকবিডি হচ্ছে বাংলাদেশী প্রফেশনাল বাংলা বই সমূহের ওয়েবসাইট। যেখান থেকে আপনি ই-বুক বিনা মূল্যে ডাউনলোড করতে পারবেন। যে বইগুলো আপনাকে আইটি আউটসোর্সিং-এ প্রফেশনাল ক্যারিয়ার গড়ে তুলতে সাহায্য করবে। এ ছাড়াও নিজে নিজে কোন প্রকার ট্রেনিং ছাড়াই যে কোন বিষয়ের উপর প্রফেশনাল দক্ষতা অর্জন করতে পারেন এই বইগুলো পড়ে। আর আপনাদের কোন পরামর্শ থাকলে অবশ্যই আমাদেরকে জানান, এই ঠিকানায় :- info@bookbd.info

১৩: মডিফায়ার (Modifiers):

বিশেষ ধরনের কিছু কী ওয়াড আমরা ব্যবহার করে থাকি ক্লাস, মেথড, ফাংশন এবং ভ্যারিয়েবল এর পূর্বে আর এটিকেই বলে modifier, অ্যাকসেসকে কন্ট্রল করার জন্য আমরা মডিফায়ার ব্যবহার করে থাকি। মডিফায়ার কে আবার visibility ও বলা হয়। modifier গুলো হচ্ছে যেমন public, private, protected, Final এবং Abstract

■ পাবলিক (Public):

কোন ক্লাস, ভেরিয়েবল এবং ফাংশন এর পূর্বে পাবলিক কী-ওয়াডটি থাকলে এটিকে আপনার স্ক্রিপ্ট এর যে কোন জায়গায় ব্যবহার করা যাবে। মনে রাখবেন ঐ ক্লাস অথবা প্রোপার্টিগুলো আপনার পিএইচপি ফাইলে সব জায়গায় ব্যবহার করা যাবে। যদি কোন ক্লাসের ভিতর মেথড অথবা ভেরিয়েবল পাবলিক, সেগুলোকে ক্লাসের ভিতরে এমনকি বাইরেও ব্যবহার করা যাবে।

```
<?php  
  
class test  
{  
    public $abc='I am defined public'; // abc variable defined by public  
}  
  
$objA = new test();  
echo $objA->abc; // accessible from outside because abc defined by public  
  
?>
```

প্রোগ্রাম p24.php: পাবলিক ভেরিয়েবল

আউটপুট: I am defined public

```
<?php  
  
class test  
{  
    public function xyz() // xyz method defined by public, xyz accessible from outside class  
    {  
        echo 'I am defined public method';  
    }  
}
```

```
$objA = new test();

$objA->xyz();//public method of the class test

?>
```

প্রোগ্রাম p25.php:পাবলিক মেথড

আউটপুট: I am defined public method

```
<?php

class test
{
    public $abc='you can access me (abc) outside of test class ';
    public $xyz;
    public function xyz()
    {
        echo 'you can access me (zyz) outside of test class ';
    }
}

$objA = new test();
echo $objA->abc;//accessible from outside
$objA->xyz();//public method of the class test

?>
```

প্রোগ্রাম p26.php:পাবলিক ভেরিয়েবল এবং মেথড

আউটপুট: you can access me (abc) outside of test class you can access me (zyz) outside of test class

■ প্রাইভেট (Private):

প্রাইভেট মেথড এবং প্রোপার্টি সমূহ শুধু মাত্র ক্লাসের ভিতরে ব্যবহার করার জন্য। প্রাইভেট ভ্যারিয়েবল এবং ফাংশনকে ক্লাসের বাইরে ব্যবহার করা যাবে না। তবে ক্লাসের মাঝে ব্যবহার করা যাবে \$this কী-ওয়ার্ড ব্যবহার করে। প্রশ্ন হল কেন ফাংশন অথবা ভ্যারিয়েবলকে প্রাইভেট করবো অর্থাৎ আপনি যদি কখনো মনে করেন আপনার ক্লাসের প্রোপার্টি ক্লাসের বাইরের কেউ ব্যবহার না করুক তখন প্রাইভেট ব্যবহার করতে হবে।

```
<?php

class test
```

```

{
    private $abc='you can access me (abc) inside of of test class ';

    public function xyz()
    {
        echo $this->abc;
    }

}
$objA = new test();
//echo $objA->abc;//accessible not possible from outside the test class
$objA->xyz();//public method of the class test

```

?>

প্রোগ্রাম p27.php:প্রাইভেট ভ্যারিয়েবল

আউটপুট: you can access me (abc) inside of of test class

```

<?php

class Car {
    private $NumberOfDoors;

    public function __construct() {
        $this->NumberOfDoors = 4;
        echo $this->NumberOfDoors;
    }

}

// instantiate
$car= new Car();
// show the property - not allowed

```

?>

প্রোগ্রাম p28.php:প্রাইভেট ভ্যারিয়েবল এবং Constructor

আউটপুট: 4

■ প্রোটেক্টেড (Protected):

প্রোটেক্টেড হচ্ছে অনেকটা পাবলিক এবং প্রাইভেট এর মাঝামাঝি। তবে প্রোটেক্টেড হচ্ছে প্রাইভেট এর মত তবে সেটিকে পিএইচপি ক্লাসে ব্যবহার করা যাবে ইনহেরিটেন্স এবং ইন্টারফেস এর ক্ষেত্রে। প্রিয় পাঠক, আমরা উপস্থাপন এবং বুঝানোর

সুবিধার্থে অনেক সময় ইংরেজি বাংলা সংমিশ্রণ করে লিখছি যা ক্ষমা সুন্দর দৃষ্টিতে দেখবেন।

```
<?php

class pc // pc is parent class
{
    protected $pr=451; // define protected to stop access outside the class
                        // but the child class can access the pr variable
}

class cc extends pc // cc is child class is so it can access the pr
{
    public function testChild()
    {
        echo $this->pr; //will work because it cc is child of pc
    }
}

$objChild = new cc();
$objChild->testChild();//work because test child will call

?>
```

প্রোগ্রাম p30.php: প্রোটেক্টেড ভ্যারিয়েবল এক্সেস

আউটপুট: 451

```
<?php

class parent
{
    protected $pr;
    public $a
    protected function testParent()
    {
        echo this is test;
    }
}

class child extends parent
{
    public function testChild()
    {
        $this->testParent(); //will work because it
    }
}
```

```
}  
$objParent = new parent();  
$objParent->testParent();//Throw error  
$objChild = new Child();  
$objChild->setChild();//work because test child will call test parent.  
  
?>
```

প্রোগ্রাম p31.php:মডিফায়ার এর ব্যবহার

```
<?php  
  
class Myclass  
  
{  
  
public $title;  
  
  
public function DispTitle()  
{  
  
echo $this -> title;  
echo "<br />";  
}  
}  
  
$obj = new Myclass();  
$obj->title = "ABCXYZ";  
  
echo $obj->title;  
  
echo "<br />";  
  
echo $obj->DispTitle();  
  
?>
```

প্রোগ্রাম p32.php:মডিফায়ার এর ব্যবহার

আউটপুট: ABCXYZ

ABCXYZ

```
<?php
class Myclass
{
    protected $title=10;
}
class Baseclass extends Myclass
{
    function __construct()
    {
        echo $this->title;
    }
}
$obj = new Baseclass();
?>
```

প্রোগ্রাম p33.php:মডিফায়ার এর ব্যবহার

আউটপুট: 10

```
<?php
class myClass
{
    public $public = 'Public';
    protected $protected = 'Protected';
    private $private = 'Private';

    function printHello()
    {
        echo $this->public;
        echo $this->protected;
        echo $this->private;
    }
}

$obj = new myClass();
echo $obj->public; // Works
echo $obj->protected; // Fatal Error
echo $obj->private; // Fatal Error

// Shows Public, Protected and Private
```

```

$obj->printHello();

class myClass2 extends myClass
{
    // We can redeclare the public and
    // protected method, but not private
    protected $protected = 'Protected2';

    function printHello()
    {
        echo $this->public;
        echo $this->protected;
        echo $this->private;
    }
}

$obj2 = new myClass2();
echo $obj->public; // Works
echo $obj2->private; // Undefined
echo $obj2->protected; // Fatal Error

// Shows Public, Protected2, not Private
$obj2->printHello();

?>

```

প্রোগ্রাম p34.php: বিভিন্ন মডিফায়ার সমূহ
আউটপুট: Public

১৪: this কী ওয়াড:

নিজ ক্লাসের কোন প্রোপার্টি যেমন ভ্যারিয়েবল এবং মেথড ব্যবহার এর জন্য this কী-ওয়াড প্রয়োজন হয়। আবার this কী-ওয়াড ব্যবহার করে অবজেক্ট দ্বারা ঐ ক্লাসের property গুলো অ্যাকসেস করা যায়।

```
<?php
```



```

class Foo
{
    public $Bar;
    public function __construct()
    {
        $this->Bar = 0; // use this to access Bar varibale
    }
    public function Baz()
    {
        $this->Bar = 10;
        echo $this->Bar;
    }
}
$f1 = new Foo();
$f2 = new Foo();
$foo1->Baz(); // sets the value of $Bar inside $foo1 to 10
               // does not affect the $Bar inside $foo2

```

?>

প্রোগ্রাম p36.php: this কী-ওয়ার্ড এর ব্যবহার
আউটপুট:

১৫: ইনহেরিটেন্স (Inheritance):

- একটি ছেলে তার বাবার সম্পদ পাবে এটাই স্বাভাবিক অর্থাৎ বাবার সম্পদ সাধারণত ছেলে বিনা অনুমতিতেই ব্যবহার করতে পারে , এটাই হচ্ছে ইনহেরিটেন্স । অর্থাৎ আপনার বাবার একটি গাড়ি আছে এটি আপনার বাবা ব্যবহার করছে আবার মাঝে মাঝে আপনি ব্যবহার করছেন কিন্তু যদি গাড়িটি আপনার চাচার হতো হয়তো বা আপনি চাইলেই ব্যবহার করতে পারতেন না, এটাই হচ্ছে Inheritance এর মজার বিষয় ।
- এখন দেখি আসলে প্রোগ্রামিং এ এটি কেমন । প্রোগ্রামিং এ একটি ক্লাস অন্য একটি ক্লাস এর ভ্যারিয়েবল অথবা মেথড ব্যবহার করতে পারবে যদি ইনহেরিটেন্স হয় । অর্থাৎ চাইল্ড ক্লাস প্যারেন্ট (parent) ক্লাস এর প্রোপার্টি ব্যবহার করতে পারবে ।

ই-কমার্স ওয়েবসাইট তৈরি করার সহজ কৌশল জানতে আমাদের লেখা ” ই-কমার্স এন্ড জুমলা! ভার্সিটি ” বইটি পড়তে পারেন ।

১৬: ইনহেরিটেন্স এর সুবিধা:

- একই কোড বার বার ব্যবহার করা যায়।
- কোড এর সাইজ অনেক কমে যায়।
- একই ধরনের কাজের জন্য একই ধরনের কোড বার বার লিখতে হয় না।
- একই ক্লাসের প্রোপার্টি অনেকগুলো ক্লাস ব্যবহার করতে পারে।
- parent ক্লাস চাইলে কিছু প্রোপার্টি hide করে রাখতে পারে অর্থাৎ চাইল্ড ক্লাস সেগুলো এ্যাকসেস (ব্যবহার) করতে পারবেনা।
- বেইস (parent) ক্লাসকে এক্সটেন্ড করা যায়।
- প্রোগ্রাম এর রান টাইম কমে যায়।
- parent ক্লাস পরিবর্তন করলে চাইল্ড ক্লাসও পরিবর্তন হয়।

১৭: ইনহেরিটেন্স প্রোগ্রাম:

ইনহেরিটেন্স তখনই হয় যখন একের অধিক ক্লাস থাকে। যেমন নিম্নে দুটি ক্লাস একটি হচ্ছে p ক্লাস যেটিকে আমরা প্যারেন্ট ক্লাস, অন্য একটি ক্লাস c যেটিকে আমরা চাইল্ড ক্লাস বলছি।

class p \longleftarrow class p এটিকে সুপার/বেইস/প্যারেন্ট ক্লাস ও বলা হয়

{

} class c এটিকে চাইল্ড/এক্সটেন্ড ক্লাস ও বলা হয়

↓
class c extends p

{

↑
এই কী-ওয়াডটি Inheritance করার জন্য ব্যবহার করতে হয়

```
}
```

প্রোগ্রাম সিনট্যাক্স ৫ : ইনহেরিটেন্স প্রোগ্রাম সিনট্যাক্স

■ ইনহেরিটেন্স করার নিয়ম হচ্ছে:

চাইল্ড ক্লাস লিখে extends কী-ওয়ার্ড লিখে সুপার ক্লাসটি লিখতে হবে।

✓ মনে রাখবেন প্যারেন্ট ক্লাস কে বেইস অথবা সুপার ক্লাসও বলা হয়।

```
<?php

class P
{
    public $n=12; // n declared in p class and assign value 12
}

class C extends P
{
    function display()
    {
        echo $this->n; // print n in the c class but c is variable in p class
                    // since c is the child of p for that reason n is printed in the display method
    }
}

$io=new C();
$io->display();
```

```
?>
```

প্রোগ্রাম p37.php: ক্লাস এবং ইনহেরিটেন্স

আউটপুট: 12

```
<?php
```

```
class class_A{
```

```

        public function hello() // hello function in the super class
        {
            return "Hello I am parent"; // hello method return Hello I am parent
        }
    }

    class my_class extends class_A
    {

        public function greeting()
        {
            echo $this->hello(); // call hello mehtod of Class_A and hello method reuturn
            'Hello I am parent'
        }
    }

    $obj = new my_class; // object of base class my_class
    echo $obj->greeting(); // call greeting method of base class

?>

```

প্রোগ্রাম p38.php: মেথড ব্যবহার এবং ইনহেরিটেন্স

আউটপুট: Hello I am parent

```

<?php

class class_A{

    public function hello($name) // hello function in the super class
    {
        return $name; // hello method return Isita
    }
}

class my_class extends class_A
{

    public function greeting($name)
    {
        echo $this->hello($name); // call hello mehtod of Class_A
    }
}

$obj = new my_class; // object of base class my_class

```

```
echo $obj->greeting('Isita'); // call greeting method of base class
```

```
?>
```

প্রোগ্রাম p39.php: ইনহেরিটেন্স এবং ভ্যালু পাসিং

আউটপুট: Isita

```
<?php
class kok
{
    var $_name;

    function Pet($name)
    {
        $this->_name = $name;
    }

    function eat()
    {
    }

    function sleep()
    {
        echo 'I am sleeping';
    }
}

class Dog extends kok
{
    function bark()
    {
    }
}

class Cat extends kok
{
    function meow()
    {
    }
}

$dog = new Dog("Max");
$dog->eat();
$dog->bark();
$dog->sleep();
```

```
$cat = new Cat("Misty");  
$cat->eat();  
$cat->meow();  
$cat->sleep();  
?>
```

প্রোগ্রাম p40.php: একের অধিক চাইল্ড ক্লাসে ইনহেরিটেন্স

আউটপুট: I am sleepingI am sleeping

```
<?php  
  
class A {  
    // code goes here  
}  
  
class B extends A {  
    // code goes here  
}  
  
class C extends B {  
    // code goes here  
}  
  
$oA = new A();  
$oB = new B();  
$oC = new C();  
  
?>
```

প্রোগ্রাম p41.php: ইনহেরিট্যান্স সিনট্যাক্স

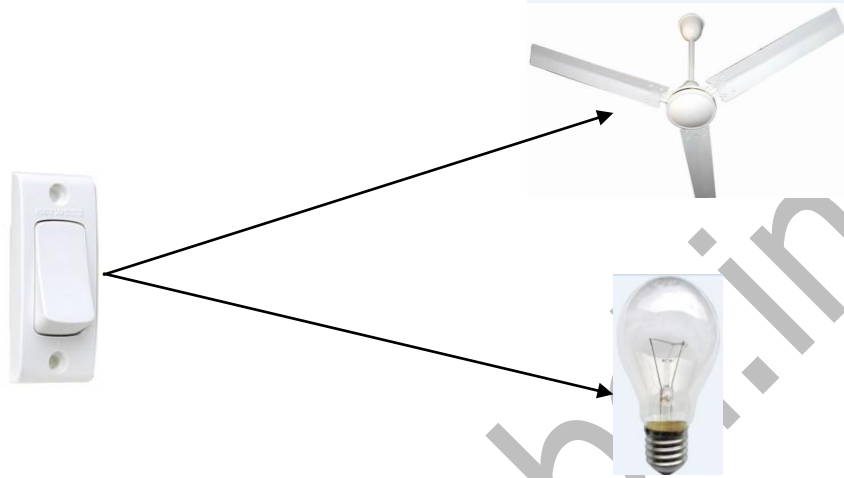
১৮: পলিমরফিজম (Polymorphism):

কারেন্ট এর সুইচ আমরা সবাই ব্যবহার করি। একই সুইচ আপনি বিভিন্ন কাজে ব্যবহার করতে পারেন যেমন ফ্যান, লাইন অন-অফ করার জন্য আর এটিই হচ্ছে পলিমরফিজম।

poly অর্থ হচ্ছে অনেক (many) আর morph অর্থ হচ্ছে ফর্ম (forms) তাহলে বলা যায়

polymorphism অর্থ হচ্ছে অনেক ফর্ম। অর্থাৎ ইলেকট্রিক সুইচ গুলো একই হলেও একেকটি আলাদা আলাদা কাজের জন্য যেমন একটি সুইচের কাজ হল লাইট অন করার জন্য অন্য সুইচের কাজ হল ফ্যান অন করার জন্য।

- প্রোগ্রামিং এ পলিমরফিজম বুঝা অনেক সহজ যেমন আপনার দুটো মেথড একই নামে আছে কিন্তু দুটো মেথড দুটো বিভিন্ন কাজ করে এটিও হচ্ছে পলিমরফিজম। অন্যভাবে বলা যায় একই ফাংশন যখন বিভিন্ন উদ্দেশ্যে ব্যবহার করা হয়।



চিত্র ১০: পলিমরফিজম এর উদাহরণ

ই-কমার্স সম্পর্কে বিস্তারিত জানতে আমাদের লেখা ই-কমার্স বইটি পড়তে পারেন।

১৯: পলিমরফিজম প্রোগ্রাম সিনট্যাক্স:

```
<?php
class A{ // parent class

function Dp(){ // Dp method in the parent class

echo "Inside the Base class<br/>";
}}

class B extends A{ // B is child class

function Dp(){ // Dp method in the child class it is also in the parent class

echo "Inside the Chlid class<br/>";
}}
// THis Dp method in both chlid and parent class this way is polymorphism
```

```
$obj=new B();
```

```
$obj->Disp();
```

```
?>
```

প্রোগ্রাম p42.php: সিনট্যাক্স পলিমরফিজম

```
<?php
```

```
class A{ // parent class
```

```
function Dp(){ // Dp method in the parent class
```

```
echo "Inside the Base class<br/>";  
}}
```

```
class B extends A{ // B is child class
```

```
function Dp(){ // Dp method in the child class it is also in the parent class
```

```
echo "Inside the Child class<br/>";  
}}
```

```
// This Dp method in both child and parent class this way is polymorphism
```

```
class C extends A
```

```
{
```

```
function Dp()
```

```
{
```

```
echo " I am in Class C";
```

```
}
```

```
}
```

```
$obj=new B();
```

```
$obj->Dp();
```

```
$oc= new C();
```

```
$oc->Dp();
```

```
?>
```

প্রোগ্রাম p43.php: পলিমরফিজম প্রোগ্রাম

আউটপুট: Inside the Child class

I am in Class C

বুকবিডি সম্পাদিত অন্যান্য বইসমূহ

- ⊙ বিগীনিং জুমলা ⊙ অ্যাডভান্সড জুমলা ⊙ প্রফেশনাল জুমলা ⊙ জুমলা টেম্পলেট মেকিং
- ⊙ বিগীনিং ওয়ার্ডপ্রেস ⊙ অ্যাডভান্সড ওয়ার্ডপ্রেস ⊙ প্রফেশনাল ওয়ার্ডপ্রেস ⊙ ইন্টারনেটে আত্মকর্মসংস্থান-১ ⊙ ইন্টারনেটে আত্মকর্মসংস্থান-২ ⊙ ই-কমার্স অ্যান্ড জুমলা! ভার্টুয়াল
- ⊙ ই-কমার্স ⊙ ম্যাক্রোমিডিয়া ড্রিমওয়েভার ⊙ সার্চ ইঞ্জিন অপটিমাইজেশন ⊙ অ্যাডভান্সড সার্চ ইঞ্জিন অপটিমাইজেশন ⊙ প্রফেশনাল সার্চ ইঞ্জিন অপটিমাইজেশন ⊙ ফরেক্স ট্রেডিং
- ⊙ ই-মার্কেটিং ⊙ অ্যাডভান্সড এইচটিএমএল ⊙ পিএইচপি অ্যান্ড মাই এসকিউএল
- ⊙ অ্যাডভান্সড পিএইচপি অ্যান্ড মাই এসকিউএল ⊙ অবজেক্ট অরিয়েন্টেড পি.এইচ.পি ডেটাবেইজ মাই এসকিউএল ⊙ সি প্রোগ্রামিং ⊙ জাভা প্রোগ্রামিং ⊙ জাভাস্ক্রিপ্ট
- ⊙ অ্যাডভান্সড ফটোশপ ⊙ অ্যাডভান্সড ইলাস্ট্রেটর ⊙ প্রফেশনাল গ্রাফিক্স ডিজাইন
- ⊙ প্রফেশনাল কম্পিউটার নেটওয়ার্কিং ⊙ ওডেস্ক এবং আউটসোর্সিং ⊙ অ্যাফিলিয়েট মার্কেটিং

www.bookbd.info

বুকবিডি হচ্ছে বাংলাদেশী প্রফেশনাল বাংলা বই সমূহের ওয়েবসাইট। যেখান থেকে আপনি ই-বুক বিনা মূল্যে ডাউনলোড করতে পারবেন। যে বইগুলো আপনাকে আইটি আউটসোর্সিং-এ প্রফেশনাল ক্যারিয়ার গড়ে তুলতে সাহায্য করবে। এ ছাড়াও নিজে নিজে কোন প্রকার ট্রেনিং ছাড়াই যে কোন বিষয়ের উপর প্রফেশনাল দক্ষতা অর্জন করতে পারেন এই বইগুলো পড়ে। আর আপনাদের কোন পরামর্শ থাকলে অবশ্যই আমাদেরকে জানানো, এই ঠিকানায় :- info@bookbd.info

২০: য়েনকেপসুলেশন (Encapsulation):

মনে করুন আপনার কাছে তিনটি জিনিস আছে যেমন প্রথমটি হচ্ছে পেন, দ্বিতীয় হচ্ছে মোবাইল আর তৃতীয়টি হচ্ছে পেন ড্রাইভ। এখন আপনি যদি চান এসব জিনিস কেউ ব্যবহার না করুক, তাহলে আপনি তা লুকিয়ে রাখবেন। অথবা ভাবলেন না, পেনটা কেউ চাইলে ব্যবহার করতে পারবে তাহলে আপনি বাকি দুটিকে লুকিয়ে রাখবেন। অর্থাৎ এই বিষয়টিকে আমরা Encapsulation ভাবে পারি।

য়েনক্যাপসুলেশন হচ্ছে এমন একটি পদ্ধতি অর্থাৎ আমার সম্পদ আমি ছাড়া আর কেউ ব্যবহার করতে পারবে কি পারবে না।

এটি অনেকটা এরকম, লুকিয়ে রেখে প্রয়োজনে কাউকে কিছু দেখতে দিলাম। য়েনক্যাপসুলেশন কে ডেটা হাইডিং ও বলা হয়। আপনি যদি মোবাইল ফোনে পাসওয়ার্ড দেন, অনেকে এটি দেখবে, নাড়াচড়া করবে কিন্তু আপনি যদি কাউকে পাসওয়ার্ড না বলেন তাহলে সে মোবাইল ওপেন করতে পারবে না।

অবশ্যই মনে রাখবেন এই বইটি পড়ার পূর্বে আমাদের লেখা "পিএইচপি অ্যান্ড মাইএসকিউল" বইটি অবশ্যই পড়বেন। অন্যথায় এই বইটি বুঝতে আপনার সমস্যা হতে পারে।

২১: প্রোগ্রামিং এ য়েনকেপসুলেশন

কোন ক্লাস এর মেথড এবং ভ্যারিয়েবলকে ঐ ক্লাস ছাড়া অন্য ক্লাস ব্যবহার করতে পারবে কি পারবে না। Encapsulation হচ্ছে এমন একটি পদ্ধতি যেটি ক্লাস এর ডেটা এবং ফাংশন এক সাথে করে রাখে ক্লাস এর ভিতরে। এটি ডেটাকে হাইড (hide) করে রাখে, যেন ক্লাস বাইর থেকে সরাসরি (Directly) ব্যবহার করতে না পারে। কিন্তু ক্লাস এর ফাংশনকে ব্যবহার করে, চাইলে ক্লাস বাইর থেকে ক্লাস এর তথ্য ব্যবহার করতে পারে।

২২: য়েনক্যাপসুলেশন প্রোগ্রামিং:

```
<?php
class enc
{
    private $p=10; // this is private varibale usually it is not possible to access outside the class
    function mn()
    {
        return $this->p; // since p is not possible use outside the class
    }
    // for this reason return it from where call the mn method
}
```

```
$obj=new enc(); // object of enc class
// using this object we cant call p variable which is in enc class
// if we write echo $obj->p; it will not work because p is private
echo $obj->mn(); // if we call mn method we get p

?>
```

প্রোগ্রাম p44.php: বেসিক সিনট্যাক্স য়েনক্যাপসুলেশন
আউটপুট: 10

```
<?php

class Hide {

    private $myname;

    function getname()

    {

        $this->myname = "damodharan";

        return $this->myname;

    }

}

class mimi {

    private $name='Badsha';

    public function sname()

    {

        echo "I am in hide class ";

        return $this->name;

    }

}

$run = new mimi();
```

```
echo $run->sname();  
$ho= new Hide();  
echo $ho->getname();
```

?>

প্রোগ্রাম p45.php: যেনক্যাপসুলেশন প্রোগ্রামিং

আউটপুট: I am in hide class Badshadamodharan

২৩: অ্যাবসট্রাকট (Abstract):

আপনার একটি পেন ড্রাইভ আছে এটি সবাই জানে এবং চাইলে নিতে পারে কিন্তু তারা ব্যবহার করতে পারে না। ধরুন আপনার মোবাইল বন্ধ নিয়েছে কল দেওয়ার জন্য কিন্তু মোবাইল সে ওপেন করতে পারছে না। কারণ আপনি মোবাইলে পাসওয়ার্ড দিয়ে রেখেছেন আর এ ব্যাপারটিই হচ্ছে অ্যাবসট্রাকট।

২৪: প্রোগ্রামিং অ্যাবসট্রাকট:

অ্যাবসট্রাকট ক্লাসকে অন্য কোন ক্লাস ইনহেরিট করে। ক্লাস এবং মেথড দুটোই Abstract হতে পারে তবে মেথড Abstract হলে ক্লাস অবশ্যই Abstract হতে হবে। মনে রাখতে হবে কোন Abstract ক্লাস এর মেথড Abstract হলে সেই মেথড এর ইমপ্লিমেন্টেশন (implementation) অর্থাৎ বডিতে কিছু থাকবে না।

- Abstract ক্লাস এর কোন অবজেক্ট create (তৈরি) করা যায় না।

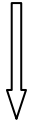
ই-কমার্স ওয়েবসাইট তৈরি করার সহজ কৌশল জানতে আমাদের লেখা "ই-কমার্স এন্ড জুমলা! ভার্সিটি" বইটি পড়তে পারেন।

- যে ক্লাস Abstract ক্লাসকে ইনহেরিট করবে সেই ক্লাসে মেথড গুলোর বডি (implementation) ডিফাইন করা থাকতে হবে

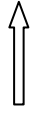
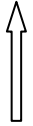
২৫: অ্যাবসট্রাকট ক্লাস সিনট্যাক্স:

নিম্নে একটি অ্যাবস্ট্রাক্ট ক্লাস এর নিয়ম দেখানো হল

abstract কী-ওয়াড



abstract class nima { }



class কী-ওয়াড ক্লাস নেইম

প্রোগ্রাম সিনট্যাক্স ৫ : অ্যাবস্ট্রাক্ট ক্লাস প্রোগ্রাম সিনট্যাক্স

- অ্যাবস্ট্রাক্ট ক্লাস এর পূর্বে অবশ্যই abstract কী-ওয়াডটি লিখতে হবে। তারপর class কী-ওয়াডটি এর পর ক্লাস এর নাম

অবশ্যই মনে রাখবেন এই বইটি পড়ার পূর্বে আমাদের লেখা "পিএইচপি অ্যান্ড মাইএসকিউল" বইটি অবশ্যই পড়বেন। অন্যথায় এই বইটি বুঝতে আপনার সমস্যা হতে পারে।

এবার দেখবো abstract মেথড লিখার নিয়ম

abstract কী-ওয়াড

ফাংশন কী-ওয়ার্ড

abstract public function myni();

সেমিকোলন

মডিফায়ার

প্রথম বন্ধনী

মেথড নেইম

প্রোগ্রাম সিনট্যাক্স ৬: অ্যাবসট্রাক্ট প্রোগ্রাম ক্লাস সিনট্যাক্স

- ✓ abstract মেথড এর ভিতর কিছু লেখা থাকবে না

```
<?php
```

```
abstract class nima
```

```
abstract public function myni();
```

```
?
```

২৬: অ্যাবসট্রাক্ট প্রোগ্রামিং:

```
<?php
```

```
abstract class Foo { // Foo is abstract class
```

```
    abstract public function foobar(); // foobar is abstract method no body of this method
```

```
}
```

```
class Baz extends Foo {
```

```

        public function foobar() { // implemetn the foobar method in the Baz class
            echo "My name is Rashid"; // only child class can implement the abstract
method
        }
    }
}
// we cant create the object of Foo class like $ob=new Foo(), not possible because abstract class

$h=new Baz();
$h->foobar();

?>

```

প্রোগ্রাম p46.php: বেসিক এ্যাবসট্যাকট প্রোগ্রাম
আউটপুট: My name is Rashid

```

<?php

abstract class A{

abstract function disp();

}

class B extends A{

public function disp(){

echo "Inside the child class<br/>";

}

}

class C extends A{

public function disp(){

echo "Inside the child class 2<br/>";

}

}

$two=new B();

echo "<b>Calling from the child class B:</b><br/>";

```

```
$two->disp();  
  
echo "<b>Calling from the child class C:</b><br/>";  
  
$three=new C();  
  
$three->disp();  
  
?>
```

প্রোগ্রাম p47.php: এ্যাবসট্র্যাকট

আউটপুট: **Calling from the child class B:**

Inside the child class

Calling from the child class C:

Inside the child class 2

২৭: ইন্টারফেইস (Interface):

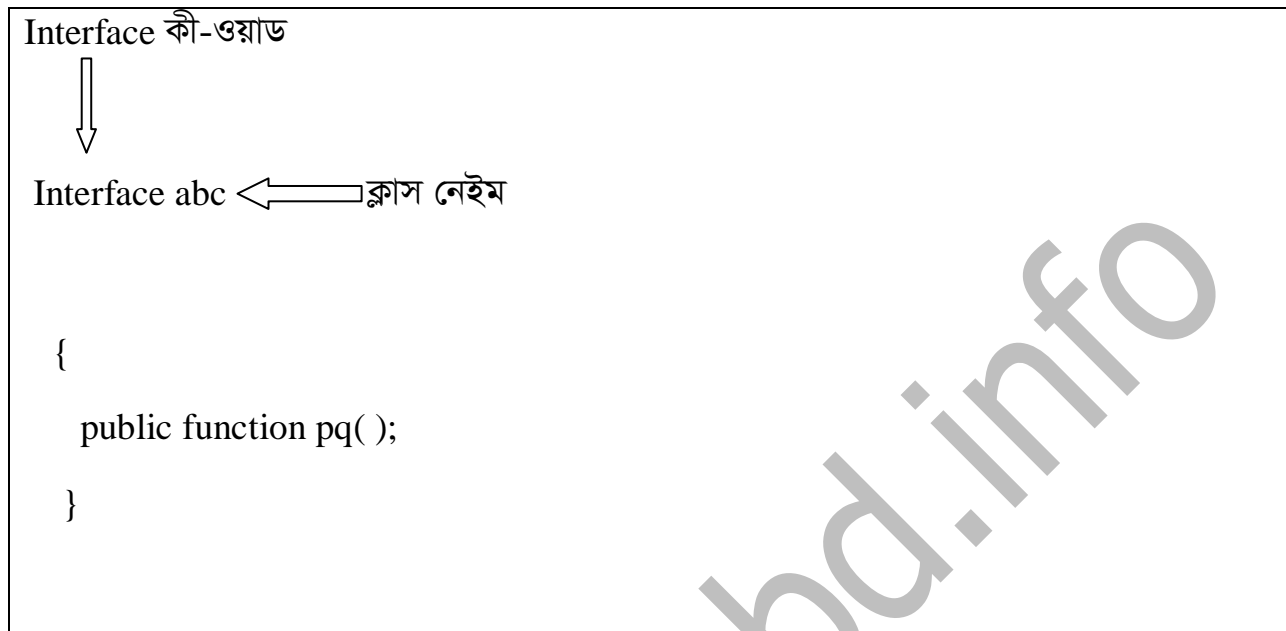
ইন্টারফেইস ক্লাসের কোন মেথড এর ডেফিনেশন থাকে না। Interface ক্লাসে শুধু মেথড এর Definition থাকে। ইন্টারফেইস ক্লাস এর মেথড কে যদি অন্য ক্লাস ব্যবহার (Implement) করতে চায়, তবে মনে রাখতে হবে ঐ ক্লাসকে অবশ্যই ইন্টারফেইস ক্লাসের সকল মেথডকে বাস্তবায়ন করতে হবে।



চিত্র ১১: রিমোট এবং টেলিভিশন

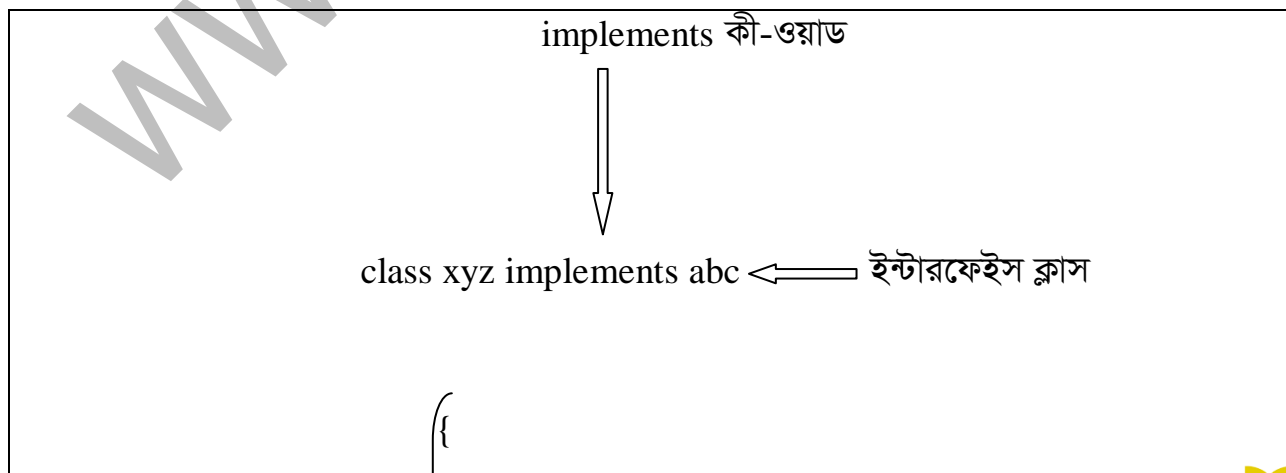
উপরের চিত্রটি লক্ষ্য করুন রিমোটটি টিভি এবং দর্শক এর মধ্যে ইন্টারফেইস হিসাবে কাজ করছে।

২৮: ইন্টারফেইস ক্লাসের সিনট্যাক্স:



প্রোগ্রাম সিনট্যাক্স ৭ : ইন্টারফেইস ক্লাস সিনট্যাক্স

- Interface হচ্ছে একটি কী-ওয়ার্ড যেটি ইন্টারফেইস ক্লাসের পূর্বে লিখতে হবে।



```

public function xyz()
{
    xyz ক্লাস
    xyz মেথড এর বডি
}
}

```

প্রোগ্রাম সিনট্যাক্স ৮ : ইন্টারফেইস ক্লাস সিনট্যাক্স

- ইন্টারফেইস ক্লাসকে যে ক্লাস ব্যবহার করবে সেই ক্লাস অবশ্যই Interface ক্লাসকে এক্সটেন্ড করতে হবে। ইন্টারফেইস ক্লাসকে এক্সটেন্ড করতে হলে অবশ্যই implements কী-ওয়ার্ডটি ব্যবহার করতে হবে।
- ইন্টারফেইস ক্লাসের মেথড অবশ্যই পাবলিক হবে এবং ভ্যারিয়েবল কনস্ট্র্যান্ট হবে।

অনলাইনের মাধ্যমে কিভাবে আয় করতে হয় বা কিভাবে আউটসোর্সিং করতে হয় তা বিস্তারিত জানতে আমাদের লেখা "ইন্টারনেটে আত্মকর্মসংস্থান-২" ও "আউটসোর্সিং এবং ওডেস্ক" বইগুলো পড়তে পারেন।

২৯: ইন্টারফেইস ক্লাস প্রোগ্রাম:

```

<?php
interface Sort {
    function getContentsAsText();
}

class Doc implements Sort {

```

```

        public function getContentsAsText() {
            return "This is Text of the Document\n";
        }

    }

    $p = new Doc();
    echo $p->getContentsAsText();

?>

```

প্রোগ্রাম p48.php: বেসিক ইন্টারফেইস ক্লাস
আউটপুট: This is Text of the Document

```

<?php

interface tem
{
    public function f1();
}

class test implements tem
{
    public function f1()
    {
        echo "Hello";
    }
}

$h = new test();
$h->f1();
?>

```

প্রোগ্রাম p49.php: ইন্টারফেইস ক্লাস
আউটপুট: Hello

```

<?php

interface employee
{
    function setdata($empname,$empage);
    function outputData();
}

```

```
}  
  
class Payment implements employee  
{  
    function setdata($empname,$empage)  
    {  
        //setdata method functionality if any  
    }  
  
    function outputData()  
    {  
        echo "Inside Payment Class";  
    }  
}  
  
$p = new Payment();  
$p->outputData();  
?>
```

প্রোগ্রাম p50.php: ইন্টারফেইস employee ক্লাস

আউটপুট: Inside Payment Class

বুকবিডি সম্পাদিত অন্যান্য বইসমূহ

- বিগীনিং জুমলা
- অ্যাডভান্সড জুমলা
- প্রফেশনাল জুমলা
- জুমলা টেমপ্লেট মেকিং
- বিগীনিং ওয়ার্ডপ্রেস
- অ্যাডভান্সড ওয়ার্ডপ্রেস
- প্রফেশনাল ওয়ার্ডপ্রেস
- ইন্টারনেটে আত্মকর্মসংস্থান-১
- ইন্টারনেটে আত্মকর্মসংস্থান-২
- ই-কমার্স অ্যান্ড জুমলা! ভার্সুয়াল
- ই-কমার্স
- ম্যাক্রোমিডিয়া ড্রিমওয়েভার
- সার্চ ইঞ্জিন অপটিমাইজেশন
- অ্যাডভান্সড সার্চ ইঞ্জিন অপটিমাইজেশন
- প্রফেশনাল সার্চ ইঞ্জিন অপটিমাইজেশন
- ফরেক্স ট্রেডিং
- ই-মার্কেটিং
- এইচ টি এম এল-৫
- অ্যাডভান্সড এইচটিএমএল
- পিএইচপি অ্যান্ড মাই এসকিউএল
- অ্যাডভান্সড পিএইচপি অ্যান্ড মাই এসকিউএল
- অবজেক্ট অরিয়েন্টেড পি.এইচ.পি
- ডেটাবেস মাই এসকিউএল
- সি প্রোগ্রামিং
- জাভা প্রোগ্রামিং
- জাভাস্ক্রিপ্ট
- জুমলা! টেমপ্লেট মেকিং
- অ্যাডভান্সড ফটোশপ
- অ্যাডভান্সড ইলাস্ট্রেটর
- প্রফেশনাল গ্রাফিক্স ডিজাইন
- প্রফেশনাল কম্পিউটার নেটওয়ার্কিং
- ওডেস্ক এবং আউটসোর্সিং
- অ্যাফিলিয়েট মার্কেটিং

www.bookbd.info

বুকবিডি হচ্ছে বাংলাদেশী প্রফেশনাল বাংলা বই সমূহের ওয়েবসাইট। যেখান থেকে আপনি ই-বুক বিনা মূল্যে ডাউনলোড করতে পারবেন। যে বইগুলো আপনাকে আইটি আউটসোর্সিং-এ প্রফেশনাল ক্যারিয়ার গড়ে তুলতে সাহায্য করবে। এ ছাড়াও নিজে নিজে কোন প্রকার ট্রেনিং ছাড়াই যে কোন বিষয়ের উপর প্রফেশনাল দক্ষতা অর্জন করতে পারেন এই বইগুলো পড়ে। আর আপনাদের কোন পরামর্শ থাকলে অবশ্যই আমাদেরকে জানান, এই ঠিকানায় :- info@bookbd.info

প্রফেশনাল দক্ষতার জন্য
বুকবিডি সিরিজের বই সমূহ...

Bookbd.info

ফ্রীল্যান্সিং এবং আউটসোর্সিং
তথ্যপ্রযুক্তি বিষয়ক
ই-বুক ডাউনলোড
www.bookbd.info

Bookbd Series...
f /Ebookbd

**THE
END**

