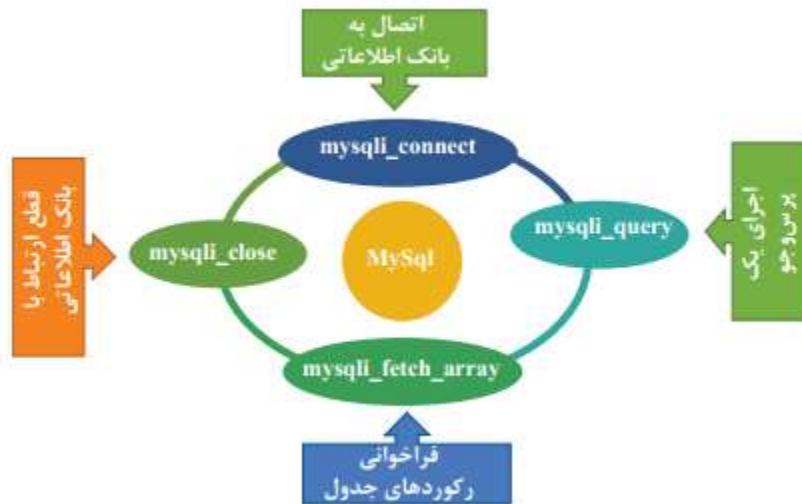


دستورات php برای کار با پایگاه داده

در زبان برنامه نویسی php دستوراتی پیش بینی شده است که برای برقراری ارتباط بین صفحات وب پویا و پایگاه داده استفاده می شود. در این بخش با دستورهایی کاربردی آشنا خواهید شد که مدیریت پایگاه داده و جداول تعریف شده در سرویس دهنده mysql را بر عهده خواهند گرفت.



تابع mysql_connect

از این تابع برای ایجاد یک اتصال به سرویس دهنده MySQL و انتخاب یک پایگاه داده استفاده می شود.

شکل استفاده:

(نام پایگاه داده, رمز عبور, نام کاربر, نام سرور) = mysql_connect = متغیر اتصال

مثال: اتصال به سرور localhost و انتخاب پایگاه داده shop_db از طریق کاربری بنام root بدون رمز عبور

`$link=mysql_connect('localhost','root','','shop_db');`

سوال: با توجه به دستور بالا، کدام گزینه نام پایگاه داده را مشخص می کند؟

الف- localhost ب- root ج- link د- shop_db

سوال: در کدام گزینه لیست پارامترهای تابع mysql_connect به ترتیب آمده است؟

الف- نام سرور، نام پایگاه داده، نام کاربر، رمز عبور ب- نام پایگاه داده، نام سرور، نام کاربر، رمز عبور

ج- نام سرور، نام کاربر، رمز عبور، نام پایگاه داده د- نام پایگاه داده، نام کاربر، رمز عبور، نام سرور

نکته: تابع فوق معادل ورود (log in) به برنامه phpMyAdmin با استفاده از کاربر root است.

تابع mysql_connect_errno

این تابع برای بررسی موفقیت آمیز بودن اتصال به سرویس دهنده MySQL استفاده می شود. خروجی تابع در

صورت عدم اتصال به سرویس دهنده MySQL، شماره خطای رخ داده است و در صورت برقراری اتصال عدد ۰

خواهد بود.

سوال: در صورت اتصال موفق به سرویس دهنده پایگاه داده، خروجی تابع mysql_connect_errno چیست؟

الف - ۱ - ب - ۰ ج - ۱ د - null

mysql_connect_error

از این تابع وقتی استفاده می شود که بخواهیم شرح خطای عدم اتصال به سرویس دهنده MySQL را نمایش دهیم. در صورت اتصال موفق خروجی این تابع null است.

سوال: در صورت اتصال موفق به سرویس دهنده پایگاه داده، خروجی تابع mysql_connect_error چیست؟

الف - ۱ - ب - ۰ ج - ۱ د - null

نکته: در دستور if، اگر بجای شرط عدد قرار گیرد، عدد صفر به معنی false و هر عددی غیر از صفر به معنی true است. بنابراین با استفاده از دستورات زیر می توان عدم اتصال به پایگاه داده را تشخیص و از ادامه ی اجرای کدهای صفحه وب جلوگیری کرد:

```
if(mysql_connect_errno())
    exit("عدم اتصال به پایگاه داده به دلیل خطای:");
```

سوال: خروجی دستور زیر چیست؟ (با فرض اینکه نام پایگاه داده را اشتباه نوشته باشیم)

الف) ۰ ب) null ج) ok د) no

```
$link=mysql_connect('localhost','root','','shop_db');
if(mysql_connect_errno())
    echo "no";
else
    echo "ok";
```

mysql_query

این تابع پرس و جوی مورد نظر یا همان فرمان های sql را روی پایگاه داده اجرا می کند. و نتیجه true به معنای اجرای درست این فرمان است. شکل استفاده:

```
mysql_query(پرس و جو, متغیر اتصال) = متغیر
```

مثال: درج اطلاعات کاربران در جدول users

```
$query="INSERT INTO users (realname, username, password, email)
VALUES ('ali mohammadi','ali','12345','ali@gmail.com)";
if(mysql_query($link,$query)===true)
    echo "اطلاعات کاربر با موفقیت ثبت شد";
```

سوال: برای اجرای پرس و جوی مقابل از کدام تابع استفاده می شود؟

“SELECT * FORM users “

الف - mysql_connect ب - mysql_query ج - mysql_close د - mysql_fetch_array


تابع `mysqli_close`

این تابع اتصال ایجادشده با سرویس دهنده MySQL را قطع می کند

شکل استفاده:

`mysqli_close(متغیر اتصال);`

نکته: تابع فوق معادل دکمه `log out` یا خروج از برنامه `phpMyAdmin` است.

سوال: دکمه `log out` () در برنامه `phpMyAdmin` معادل کدام تابع است؟

الف- `mysqli_connect` ب- `mysqli_fetch_array` ج- `mysqli_query` د- `mysqli_close`

تابع `mysqli_fetch_array`

این تابع محتوای یک سطر از جدول را در یک آرایه ذخیره می کند. اندیس های این آرایه اسامی فیلدهای جدولی است که پرس وجو برای آن انجام شده است. همچنین برای دسترسی به عناصر آرایه می توان از اندیس عددی نیز استفاده کرد که ترتیب این اندیس ها بر اساس موقعیت فیلدها در جدول تعیین می شود. خروجی این تابع یک آرایه است. هنگام استفاده از این تابع می توان خروجی را در یک متغیر ذخیره کرد.

شکل استفاده:

`(خروجی تابع اجرا کننده پرس و جو) = mysqli_fetch_array(نام آرایه);`

مثال: اطلاعات جدول `users` را نمایش دهید

```
$query="SELECT * FROM users ";
$result=mysqli_query($link,$q);
while($row= mysqli_fetch_array($result))
    echo "$row['realname']". "$row['username']". "$row['password']". "$row['email']". "<br/>";
```

فعالیت کارگاهی - کد مربوط به عضویت در سایت از طریق فرم زیر:

<input type="text"/>	نام واقعی
<input type="text"/>	نام کاربری
<input type="text"/>	کلمه عبور
<input type="text"/>	تکرار کلمه عبور
<input type="text"/>	پست الکترونیکی
<input type="button" value="ثبت نام"/> <input type="button" value="جدید"/>	

```

$link=mysqli_connect('localhost','root','','shop_db');
if(mysqli_connect_errno())
    exit("خطای زیر رخ داده است");
$query1="SELECT * FROM users WHERE username='$username'";
$result=mysqli_query($link,$query1);
if($row=mysqli_fetch_array($result))
    exit ("نام کاربری تکراری است");
$query2="INSERT INTO users
VALUES ('$realname','$username','$password','$email','0')";
if(mysqli_query($link,$query2)===true)
    echo "عضویت شما با موفقیت انجام شد";
else
    echo "عضویت انجام نشد";
mysqli_close($link);
    
```

جلسه (session)

آرایه \$_SESSION

session یک پرونده در پوشه موقت روی سرور ایجاد می کند، که متغیرهای ثبت شده در session و مقادیر آنها در آن پرونده نگهداری و ذخیره می شوند. یکی از روش های در دسترس قرار دادن این اطلاعات در صفحات مختلف تارنما بهره گیری از آرایه \$_SESSION است.

تعریف متغیر و مقداردهی آن در session

مقدار = ["نام متغیر"] \$_SESSION

تابع session_start

برای اینکه بتوان از آرایه \$_SESSION استفاده کرد باید تابع session_start را فراخوانی کرد. قبل از فراخوانی تابع session_start نباید دسترسی از جنس HTML یا مشابه آن وجود داشته باشد. در صورتی که قبل از فراخوانی این تابع مثلاً با echo عبارتی را نمایش داده باشید با خطا و عدم کارکرد صحیح session مواجه خواهید شد.

در هر یک از صفحات وب که می خواهید از متغیرهای session استفاده کنید، باید در ابتدای صفحه وب مورد نظر تابع `session_start` را فراخوانی کنید. به سادگی با استفاده از تابع `isset` می توانید وجود یا عدم وجود یک متغیر در `session` را بررسی کنید.

تابع `unset`

برای حذف محتوای یک متغیر از تابع `unset` استفاده می شود.

شکل کلی تابع `unset`

`unset` (نام متغیر);

تابع `session_unset`

برای حذف همه متغیرهای تعریف شده در `session` استفاده می شود. این تابع آرگومان ورودی ندارد.

تابع `session_destroy`

برای از بین بردن و حذف کامل پرونده ایجاد شده برای `session` در وب سرور استفاده می شود. با استفاده از این تابع همه متغیرهای تعریف شده در `session` نیز حذف خواهند شد. این تابع آرگومان ورودی ندارد.

سوال: برای حذف پرونده‌ی مربوط به `session` از کدام تابع استفاده می شود؟

الف - `isset` ب - `unset` ج - `session_unset` د - `session_destroy`

سوال: برای حذف متغیرهای `session` از کدام تابع نمی توان استفاده کرد؟

الف - `isset` ب - `unset` ج - `session_unset` د - `session_destroy`

مثال: محاسبه ی تعداد بازدید از سایت

```
session_start();
if(isset($_SESSION['$counter']))
    $_SESSION['$counter']++;
else
    $_SESSION['$counter']=1;
echo "تعداد بازدید: ".$_SESSION['$counter'];
```

فعالیت کارگاهی: ورود به سایت از طریق فرم زیر (تعیین وضعیت ورود و تشخیص نوع کاربر)

<input type="text"/>	نام کاربری
<input type="text"/>	کلمه عبور
<input type="button" value="جدید"/>	<input type="button" value="ورود"/>

```

$link=mysqli_connect('localhost','root','','shop_db');
if(mysqli_connect_errno())
    exit("خطای زیر رخ داده است");
$query="SELECT * FROM users WHERE username='$username' and password='$password'";
$result=mysqli_query($link,$query);
if($row=mysqli_fetch_array($result))
{
    $_SESSION['state_login']=true; // وضعیت ورود کاربر
    $_SESSION['realname']=$row['realname']; // نام واقعی
    if($row['type']==0)
        $_SESSION['user_type']="public"; // کاربر عمومی
    else if ($row['type']==1)
    {
        $_SESSION['user_type']="admin"; // کاربر مدیر
    }
}
else
    echo " نام کاربری یا رمز عبور اشتباه است ";
mysqli_close($link);

```

توابع مربوط به ارسال پرونده به سمت سرور

تابع basename

این تابع مسیر کامل یک پرونده را از ورودی دریافت کرده، در خروجی نام و پسوند پرونده را به صورت یک رشته برمی گرداند.

شکل کلی تابع basename

؛ (مسیر کامل یک پرونده و نام آن به صورت رشته ای) basename = نام متغیر

مثال: خروجی دستور زیر چیست؟

```
echo basename("wamp\www\index.php"); // index.php
```

خروجی دستور

سوال: خروجی دستور basename("ets/php.ini") چیست؟

الف) ini ب) php ج) php.ini د) etc

تابع file_exists

این تابع مسیر کامل یک پرونده یا یک پوشه را از ورودی دریافت کرده، در خروجی وجود یا عدم وجود آن را به صورت یک مقدار منطقی true یا false برمی گرداند.

شکل کلی تابع file_exists

؛ (مسیر کامل یک پرونده یا پوشه و نام آن به صورت رشته ای) file_exists = نام متغیر

مثال: بررسی کنید پرونده index.php در مسیر wamp\www\index.php وجود دارد یا خیر؟

```
$filename=" wamp\www\index.php";
if(file_exist($filename))
    echo "در حافظه وجود دارد ".$filename." پرونده";
else
    echo "در حافظه وجود ندارد ".$filename." پرونده";
```

سوال: وجود یا عدم وجود یک پرونده روی حافظه، با کدام تابع تشخیص داده می شود؟

الف) basename ب) pathinfo ج) file_exist د) getimagesize

تابع pathinfo

این تابع مسیر کامل یک پرونده و یک شناسه ثابت را از ورودی دریافت کرده، در خروجی یک آرایه انجمنی از اطلاعات پرونده و مسیر دسترسی به آن را برمی گرداند

شکل کلی تابع pathinfo

(نام شناسه، مسیر کامل یک پرونده و نام آن به صورت رشته‌ای) pathinfo = نام آرایه

شناسه‌های تابع pathinfo

نام شناسه	توضیحات شناسه
PATHINFO_DIRNAME	مسیر و نام پوشه‌های پرونده را برمی گرداند.
PATHINFO_BASENAME	نام و پسوند پرونده را برمی گرداند.
PATHINFO_EXTENSION	پسوند پرونده را برمی گرداند.
PATHINFO_FILENAME	نام را بدون پسوند پرونده برمی گرداند.

مثال: به خروجی دستورات زیر دقت کنید

```
$filename=" wamp\www\index.php";
echo pathinfo($filename,PATHINFO_DIRNAME)."<br/>"; // wamp\www
echo pathinfo($filename,PATHINFO_BASENAME)."<br/>"; // index.php
echo pathinfo($filename,PATHINFO_EXTENSION)."<br/>"; // php
echo pathinfo($filename,PATHINFO_FILENAME)."<br/>"; // index
```

سوال: در تابع pathinfo شناسه مربوط به پسوند پرونده کدام است؟

- الف) PATHINFO_DIRNAME ب) PATHINFO_BASENAME
ج) PATHINFO_FILENAME د) PATHINFO_EXTENSION

تابع getimagesize

این تابع در ورودی مسیر کامل و نام یک تصویر را دریافت کرده، در خروجی به صورت یک آرایه انجمنی اطلاعات کاملی از تصویر را برمی گرداند. در صورتی که پرونده ورودی از نوع تصویری نباشد، خروجی این تابع false خواهد بود

شکل کلی تابع getimagesize

; (مسیر کامل و نام یک پرونده تصویری به صورت رشته ای) getimagesize = نام آرایه

اندیس های آرایه انجمنی خروجی تابع getimagesize

اندیس انجمنی آرایه در خروجی	توضیحات	مثال در خروجی
۰	عرض تصویر بر اساس پیکسل	275
۱	ارتفاع تصویر بر اساس پیکسل	183
۳	عرض و ارتفاع تصویر به صورت رشته ای	width="275" height="183"
bits	تعداد بیت های رنگ تصویر	8
channels	تعداد کانال های رنگ تصویر	3
mime	نوع تصویر	image/jpeg

مثال: به خروجی دستور زیر دقت کنید (ابعاد تصویر flag.jpg بر اساس پیکسل ۲۷۵×۱۸۳ است)

```
$info=getimagesize("img\flag.jpg");
echo $info[3]; //width="275" height="183"
echo $info['mime']; //image/jpeg
```

سوال: خروجی دستور getimagesize("file1.txt") چیست؟ (پرونده انتخابی تصویر نیست.)

الف- true ب- false ج- 0 د- 1

سوال: با توجه به قطعه کد زیر، با کدام گزینه می توان به نوع تصویر دست یافت؟

```
$info=getimagesize("img\flag.jpg");
echo $info[0];      ب- $info[1]      ج- $info[3]      د- $info['mime']
```

آرایه \$_FILES

\$_FILES یک آرایه انجمنی است که اطلاعات پرونده را نگهداری می کند. نام یک شی از نوع برچسب **file** (نام پرونده) و یک شناسه ثابت، کلیدهای این آرایه هستند. محتوای عناصر این آرایه اطلاعات پرونده دریافتی است

روش استفاده از آرایه \$_FILES

```
$_FILES["نام شناسه"]["نام برچسب از نوع file"];
```

شناسه های آرایه \$_FILES

نام شناسه	توضیحات شناسه	مثال
name	نام و پسوند پرونده ورودی را برمی گرداند.	images6.jpg
type	نوع پرونده ورودی را برمی گرداند.	image/jpeg
tmp_name	مسیر محلی و نام پرونده ورودی را برمی گرداند که در سیستم کاربر به صورت موقت با نامی جدید ذخیره شده است.	C:\Users\medu\AppData\Local\Temp\php360F.tmp
error	شماره خطای ایجاد شده را برمی گرداند.	0
size	اندازه و حجم پرونده ورودی را بر حسب بایت برمی گرداند.	14763

مثال: با فرض ارسال یک پرونده از طریق فرم ، می توان به صورت زیر به جزئیات پرونده دسترسی پیدا کرد.

```
echo $_FILES['filefield']['name'].<br/>;
```

```
echo $_FILES['filefield']['type'].<br/>;
```

```
echo $_FILES['filefield']['tmp_name'].<br/>;
```

```
echo $_FILES['filefield']['size'].<br/>;
```

```
echo $_FILES['filefield']['error'].<br/>;
```

تابع `move_uploaded_file`

این تابع یک پرونده را از مسیر مبدأ به مسیر مقصد انتقال می دهد. ورودی این تابع مسیر و نام پرونده در مبدأ و مسیر و نام پرونده در مقصد است. خروجی تابع در صورت موفقیت آمیز بودن انتقال پرونده به مقصد `true` و در صورت عدم موفقیت `false` است. در صورتی که پرونده ای هم نام در مقصد وجود داشته باشد، پرونده جدید جایگزین خواهد شد. این تابع فقط پرونده هایی را می تواند به مقصد منتقل کند که از طریق `POST` در یک نمون برگ ارسال شده باشند.

شکل کلی تابع `move_uploaded_file`

؛ (مسیر و نام پرونده در مقصد ، مسیر و نام پرونده در مبدأ) `move_uploaded_file` = نام متغیر

مثال: به دستورات زیر دقت کنید

`$sourcefile=$_FILES['filefield']['tmp_name'];//` تصویری که توسط فرم ارسال شده است.

`$targetfile="images/".$_FILES['filefield']['name'];//` مقصد پرونده سمت سرور

`if(move_uploaded_file($sourcefile, $targetfile))`

`echo "پرونده با موفقیت ارسال شد";`

`else`

`echo "خطا در ارسال پرونده!!!!";`

سوال: هنگام ارسال پرونده توسط تابع `move_uploaded_file`، در صورتی که پرونده ای هم نام در مقصد وجود داشته باشد،

نادرست

درست

پرونده جدید جایگزین خواهد شد.

سوال: تابع `move_uploaded_file`، فقط پرونده هایی را می تواند به مقصد منتقل کند که از طریق ارسال شده

باشند. (GET / POST)

فعالیت کارگاهی: ثبت محصول از طریق فرم زیر (اعتبارسنجی پرونده ارسالی)

فرم ثبت محصول جدید

<input style="width: 95%;" type="text"/>	*کد کالا:
<input style="width: 95%;" type="text"/>	*نام کالا:
<input style="width: 95%;" type="text"/>	*تعداد موجودی:
<input style="width: 95%;" type="text"/>	*قیمت کالا:
No file selected. <input type="button" value="...Browse"/>	*آپلود تصویر کالا:
<input style="width: 95%; height: 100%;" type="text"/>	*توضیحات تکمیلی کالا:
<input type="button" value="جدید"/>	<input type="button" value="ثبت"/>

```

if(isset($_POST['code']) && !empty($_POST['code']) &&
isset($_POST['name']) && !empty($_POST['name']) &&
isset($_POST['qty']) && !empty($_POST['qty']) &&
isset($_POST['price']) && !empty($_POST['price']) &&
isset($_POST['detail'])&& !empty($_POST['detail']) &&
$_FILES['image']['name']!="")// پرونده انتخاب شده است
{
    $code=$_POST['code'];
    $name=$_POST['name'];
    $qty=$_POST['qty'];
    $price=$_POST['price'];
    $image=$_FILES["image"]["name"];
    $detail=$_POST['detail'];
}
else
    exit ("اطلاعات را بطور کامل ثبت کنید");
$uploadOk=1;
$sourceFile=$_FILES["image"]["tmp_name"]; // مسیر محلی تصویر انتخاب شده
$targetFile="images/products/" .$_FILES["image"]["name"]; // مسیر مقصد پرونده ارسالی روی سرور
$fileSize=$_FILES["image"]["size"]/1024; // تعیین اندازه تصویر بر حسب کیلوبایت
$fileExtension=pathinfo($targetFile,PATHINFO_EXTENSION); // جدا کردن پسوند تصویر
$fileImage=getimagesize($sourceFile);
if($fileImage)
{
    
```

```
echo "<br/>";
echo " :$fileExtension";
echo " :$fileSize";
    $uploadOk=1;
}
else
{
    echo "<br/>";
    $uploadOk=0;
}
if(file_exists($targetFile))
{
    echo "<br/>";
    $uploadOk=0;
}
if($fileSize>500)
{
    echo "<br/>";
    $uploadOk=0;
}
if($fileExtension=="jpg" || $fileExtension=="png" || $fileExtension=="gif")
{
    echo "<br/>";
    $uploadOk=1;
}
else
{
    echo "<br/>";
    $uploadOk=0;
}
if($uploadOk==1)
{
    if(move_uploaded_file($sourceFile,$targetFile)===true)
    {
        echo "<br/>";
    }
}
```

```
        $uploadOk=1;
    }
    else
    {
        echo "خطا در ارسال پرونده "<br/>";
        $uploadOk=0;
    }
}
else
    echo "پرونده ارسال نشد "<br/>";
if($uploadOk==1)
{
    $link=mysqli_connect('localhost','root','','shop_db');
    if(mysqli_connect_errno())
    {
        echo "خطای اتصال به پایگاه داده "<br/>".mysqli_connect_error();
        exit();
    }
    $query="INSERT INTO products
            VALUES ('$code','$name','$qty','$price','$image','$detail)";
    if(mysqli_query($link,$query)===true)
        echo "کالا با موفقیت به انبار اضافه شد ";
    else
        echo "خطا در ثبت اطلاعات در پایگاه داده ";
    mysqli_close($link);
}
```

توابع کار با رشته

تابع substr

این تابع بخشی از یک رشته را بر می گرداند.

شکل کلی تابع substr

(تعداد نویسه‌ها، اندیس شروع اولین نویسه، رشته یا متغیر رشته‌ای) substr = نام متغیر

- اندیس اولین نویسه ۰ است.

- نوشتن تعداد نویسه‌ها اجباری نیست.

- نوشتن اعداد منفی برای اندیس شروع اولین نویسه باعث می شود بررسی رشته به صورت معکوس انجام شود.

مثال: به خروجی دستورات زیر دقت کنید

```
echo substr("abcdef",1,3); // bcd
echo substr("abcdef",1); // bcdef
echo substr("abcdef",0,4); // abcd
echo substr("abcdef",0,8); // abcdef
echo substr("abcdef",-1,1); // f
echo substr("abcdef",-3,2); // de
```

فعالیت کارگاهی: نمایش محصولات فروشگاه در صفحه index.php بصورت یک جدول سه ستونی

```

$link=mysqli_connect('localhost','root','','shop_db');
if(mysqli_connect_errno())
    exit ("خطای اتصال به پایگاه داده ".mysqli_connect_error());
$query="SELECT * FROM products";
$result=mysqli_query($link,$query);
?>
<table style="width:100%;border:1px;"><tr>
<?php
$counter=0;
while($row=mysqli_fetch_array($result))
{
    $counter++;
?>
<td style="width:33%;border-style:dotted;vertical-align:top;">
    <h4 style="color:brown;" ><?php echo $row['pro_name']; ?></h4>
    <br/>
    قیمت: <?php echo $row['pro_price']; ?> &nbsp;ریال <br/>
    تعداد موجودی: <span style="color:red;"> <?php echo $row['pro_qty']; ?> </span><br/>
    توضیحات: <span style="color:green;"> <?php echo $row['pro_detail']; ?> ...
    </span><br/><br/>
    <b><a href="product_detail.php?id=<?php echo $row['pro_code'];?>" style="text-decoration:none;">خرید و تکمیلی توضیحات</a></b><br/>
</td>
<?php
if($counter%3==0)
    echo"</tr><tr>";
}
?>
</tr></table>
<?php
mysqli_close($link);
?>

```

تذکر: بخش فعالیت های کارگاهی برای امتحان عملی در نظر گرفته شده است و در امتحان کتبی سوال طرح نمی شود.