

Amey Thakur

B - 50

Amey Thakur B-50

**Terna Engineering College
Department of Computer Engineering**

Assignment No. 2

Class: SE – A, B, C

Year: FH - 20

Subject: OSTL

Sr. No.	Question	CO Mapping	PO Mapping	BL
1	Explain the following: 1. Graphical User Interface in Python 2. Networking in Python 3. Database Connectivity in Python	4		
2	Discuss on Django framework for creating Web application.	4		
3	Discuss Features of Perl.	4		
4	Elaborate the following: 1. Control Statements in Perl 2. Object, Package, Modules in Perl	5		
5	Explain the following: 1. Networking in Perl 2. Perl Database Connectivity	6		
6	Write a Program for file operations in Perl.	6		

Terna Engineering College, Navi Mumbai

Q.I. Explain

A) Graphical User Interface

- Python provides several different options for writing GUI based programs.

• Tkinter:

It is easiest to start with, Tkinter is Python's standard GUI (Graphical User Interface) package.

It is the most commonly used toolkit for GUI programming in python.

• JPython:

It is the python platform for Java that is providing python scripts seamless access to JAVA class libraries for the local machine.

• WxPython:

It is an open source, cross platform GUI toolkit written in C++. It is one of the alternatives to Tkinter which is bundled with python.

- There are many other interfaces available for GUI. But these are the most commonly used ones.

Standard attributes for GUI

- Dimensions

- Fonts

- Colors

- Cursors

- Anchors

- Bitmaps

Amey

Thakur

B - 50

Terna Engineering College, Navi Mumbai

Methods for geometry management.

- The pack(): This method manages the geometry of widgets in blocks.
- The grid(): This method organizes widgets in a tabular structure.
- The place(): This method organizes the widgets to place them in a specific position.

Terna Engineering College, Navi Mumbai

B] Networking in Python

- Python plays an essential role in network programming. The standard library of python has full support for network protocols, encoding and decoding of data and other networking concepts and it is simpler to write network programs in python than that of C++.

Python Network Services

- There are two levels of network service access in python

① Low - Level Access

② High - Level Access

- In the first case, programmers can use and access the basic socket support for the operating system using python's libraries. and programmers can implement both connection less and connection oriented protocols for programming
- Application level network protocols can also be accessed using high level access provided by Python libraries. These protocols are HTTP, FTP, etc.

Socket

- A socket is the end point in a flow of communication between two programs or communication channels operating over a network. They are created using a set of programming requests called socket API (Application Programming Interface).

Amey Thakur B - 50

Terna Engineering College, Navi Mumbai

Python's socket library offers classes for handling common transports as a generic interface.

- Sockets use protocols for determining the connection type for port to port communication between client and server machines.

The protocols are used for.

→ Domain Name Servers (DNS)

→ IP Addressing

→ E-mail

→ FTP (File Transfer Protocol), etc.

Socket Syntax:

`g = socket.socket(socket_family, type_of_socket, protocol = value)`

Terna Engineering College, Navi Mumbai

c) Database Connectivity

- Steps to connect a python application to database.
- ① Import mysql.connector module.
 - ② Create the connection object.
 - ③ Create the cursor object.
 - ④ Execute the query.

Creating the connection

- To create a connection between the MySQL database and the python application, the connect() method of mysql.connector module is used.
- Pass the database details like HostName, username and the database password in the method call. The method returns the connection object.
- Syntax to use the connect()

```
Connection - Object = mysql.connector.connect  
( host = <host-name>, user = <username>,  
password = <password> )
```

- Example -

```
import mysql.connector  
# Create the connection object.  
myconn = mysql.connector.connect(host = "localhost",  
user = "root", password = "google")  
# Printing the connection object.  
print(myconn)
```

Output:

```
<mysql.connector.connection.MySQLConnection object  
at 0x7fb142edd780>
```

Terna Engineering College, Navi Mumbai

Q2 Django framework for creating web application

- Django is widely used python web application framework with a "batteries included" philosophy. The principle behind batteries included is that the common functionality for building web applications should come with framework instead of a separate batteries.
- Django's primary goal to ease the creation of complex database-driven website. The framework emphasizes reusability and plugability of component, less code, low coupling, rapid development and principle of don't repeat yourself.
- Python is used throughout even for setting files and data models.
- Django also provides an optional administrative create, read, update and delete interface that is generated dynamically through introspection and configured via admin model.
- Some well known sites that use Django includes the public broadcasting service, Instagram, mozilla.
- Features: Included in to core framework are.
 - A lightweight and standalone webserver for development and testing
 - A form serialization and validation system that can translate between HTML forms and values suitable for storage and database
 - A template system that utilized the concept of inheritance borrowed from object oriented programs

Amey

Thakur

B - 50

Terna Engineering College, Navi Mumbai

- Support for middleware classes that can intervene at various stages of request processing and carry out custom fashion.
- An internationalization system including translation of Django's own component into a variety of languages.
- A serialization system that can produce and read XML and/or JSON representation of Django model instances.
- An interface to Python's built-in unit test framework.
- Django Test framework is a powerful and flexible toolkit for building web API.

Terna Engineering College, Navi Mumbai

Q.3. Features of Perl.

- It has a very simple object oriented programming syntax.
- It is easily extendible as it supports 25,000 open source modules.
- It supports unicode.
- It includes powerful tools to process text to make it compatible with markup languages like HTML, XML.
- It supports third party database including Oracle, MySQL and many others.
- It is embeddable in other systems such as web servers and database servers.
- It is open source software licensed under GNU.
- Many frameworks are written in perl.
- It can handle encrypted web data including e-commerce transactions.
- It is a cross platform language.
- It offers a regular expression engine which is able to transform any type of text.

Terna Engineering College, Navi Mumbai

Q4. Elaborate

A) Control statement in perl.

- Perl is a general purpose programming language originally developed for text manipulation and now used for a wide range of tasks including system administration, web-development, Network programming, GUI development, etc.
- Control statement change the execution from its normal sequence, when execution leaves a scope all automatic objects that were created in that scope destroyed.
- Perl supports the following control statement

① Next Statement.

- It causes the loop to skip the remainder of its body and immediately retest its condition prior to reiterating.
- We can provide a LABEL with next statement where LABEL is the label for a loop.
- Syntax.

next [LABEL]

- Example.

\$a = 0;

OUTER : while (\$a < 4) {

 \$b = 0

 print "Value of a: \$a\n"

INNER: while (\$b < 4) {

 if (\$a == 2) {

 \$a = \$a + 1;

 # jump to outer loop

 } next OUTER;

 \$b = \$b + 1;

 print "Value of b: \$b\n";

}

Terna Engineering College, Navi Mumbai

```
print "In";
f a = $a + 1;
3
```

Output:

value of a : 0

value of b : 1

value of b : 2

value of b : 3

value of b : 4

value of a : 1

value of b : 1

value of b : 2

value of b : 3

value of b : 4

value of a : 2

value of a : 3

value of b : 1

value of b : 2

value of b : 3

value of b : 4

Terna Engineering College, Navi Mumbai

② Last statement

- Terminates the loop statement and transfers execution to the statement immediately following the loop.

- Syntax:

Last [LABEL]

- Example:

```
$a = 10;
while ($a < 20) {
    if ($a == 15) {
        # terminate the loop
        $a = $a + 1;
    }
}
```

```
print "value of a: $a\n";
$a = $a + 1;
```

}

Output:

```
value of a : 10
value of a : 11
value of a : 12
value of a : 13
value of a : 14
```

Terna Engineering College, Navi Mumbai

③ Continue Statement:

- A continue block, it is always executed before the conditional is about to be evaluated again
- syntax :

```
continue {  
    statements  
}
```

- Example

```
$a = 0;  
while ($a < 3) {  
    print "value of a = $a \n";  
    continue;  
    $a = $a + 1;  
}
```

Output:

```
value of a: 0  
value of a: 1  
value of a: 2
```

Terna Engineering College, Navi Mumbai

(4) redo statement

- Restart the loop block without evaluating the conditional again.

The continue block if any is not executed.

- Syntax:

`redo [LABEL]`

- Example:

```
$a = 0;
```

```
while ($a < 10) {
```

```
    if ($a == 5) {
```

```
        $a = $a + 1;
```

```
        redo;
```

```
}
```

```
    print "value of a = $a \n";
```

```
}
```

```
    $a = $a + 1;
```

```
}
```

Output:

Value of a = 0

value of a = 1

value of a = 2

value of a = 3

value of a = 4

value of a = 6

value of a = 7

value of a = 8

value of a = 9

Terna Engineering College, Navi Mumbai

⑤ Goto statement

- Perl supports a goto command with three forms: goto label, goto expr, goto &name.

→ goto label

- It jumps to statement labeled with LABEL and resume execution from there.

- Syntax -

goto LABEL

→ goto Expr.

- It is a generalization of goto LABEL. It expects the expression to return a label name and jumps to that labeled name.

- Syntax:

goto EXPR

→ goto &name

- It substitutes a call to the named subroutine for currently running subroutine.

- Syntax:

goto &Name

Example:

\$a = 10

Loop: do {

if (\$a == 15) {

Skip the iteration

\$a = \$a + 1;

Use goto label form

} goto loop;

Terna Engineering College, Navi Mumbai

```
print " value of a = $a \n";  
$a = $a + 1;  
3 while ($a < 10);
```

Output:

value of a = 10

value of a = 11

value of a = 12

value of a = 13

value of a = 14

value of a = 15

value of a = 16

value of a = 17

value of a = 18

value of a = 19

Terna Engineering College, Navi Mumbai

B) Object, Packages, Module in Perl

- A perl package is a collection of code which resides in its own namespace. Perl module is a package defined in a file having the same name as the package and having extension .pm.
- Two different modules may contain a variable or a function of the same name.
- Any variable which is not contained in any packages belongs to main package.
- Therefore, all the variables being used, belongs to main package with the declaration of additional packages, it is maintained that variables in different packages do not interfere with each other.

Package statement

- A package statement switches the current naming context to a specified namespace (symbol table)
- If the named package does not exist, new namespace is first created.
- Package stays in effect until either another package statement is invoked or until the end of current block or file.

Perl Modules

- A perl module is a reusable package defined in a library file whose name is same as the name of the package (with .pm in end).

Terna Engineering College, Navi Mumbai

Example:

```
Package FOO;
sub bar {
    print "Hello $_[0]\n";
}
sub blat {
    print "World $_[0]\n";
}
1;
```

Few notable points about module

- Functions require and use will load a module
- both use the 1st of search path in **(@INC)** to find module (you may modify it)
- Both call the eval function to process the code
- The 1; at the bottom causes to evaluate to TRUE and thus not fail.

Require function

- A module can be loaded by calling the require function

```
require FOO;
```

```
FOO::bar("a");
```

```
FOO::blat("b");
```

Here the subroutine names must be fully qualified

Use function:

- A module can be loaded by calling use function.

e.g.

```
use FOO;
```

```
bar("a");
```

```
blat("b");
```

Terna Engineering College, Navi Mumbai

- Use function will export a list symbol form of module given a few added statements inside a module.

require Exporter;

@ISA = qw (Exporter);

- then provide a list of symbols (Scalars, lists, hashes, subroutines, etc) by filling list variables

@Export

e.g:

Package MODULE;

require Exporter;

@ISA = qw (bar blat);

```
Sub bar { print "Hello $-[0]\n" }
```

```
Sub blat { print "World $-[0]\n" }
```

```
Sub blat { print "Not $-[0]\n" } # Not Exported
```

```
;
```

- Object within perl is merely a reference to data type that knows what class it belongs to.

The object is stored as a reference to object, the same scalar can hold different objects in different classes.

- To create an instance of class (an object) we need an object constructor.

- This constructor is a method within a package, we can share any kind of perl variable as an object in perl.

Aney

Thakur B - 50

Terna Engineering College, Navi Mumbai

- When creating an object in perl, we need a constructor which is a subroutine in a package that returns an object reference.

- e.g.

Package person;

sub new {

 my \$class = shift;

 my \$self = {

 first_name => shift,

 last_name => shift,

 ssn => shift,

 };

print all the values just for clarification.

print "First name is \$self->{first_name}\n";

print "Last name is \$self->{last_name}\n";

print "SSN is \$self->{ssn}\n";

bless \$self -> \$class;

return \$self;

}

\$object = new Person ("Aney", "Thakur", 23234445);

Terna Engineering College, Navi Mumbai

Q5. Explain

a) Networking in Perl.

- Socket programming in perl is a way of connecting 2 nodes on a network to communicate with each other.
- It is a one way client server setup where a client connects, sends messages to the server and server shows them using socket connection.
- One socket listens on a particular port of an IP, while other socket reaches out to the other to form a connection. The server forms the listener socket while client reaches out to server.
- Therefore, a socket is an IP address, and a port together which enables a network connection to send & receive information to other networks or system.

Stages for server side programming

- Create a socket using the `socket()` call function
Perl provides a predefined module `socket.ppm` which needs to be included in the code.
eg - use `socket`;
- `bind()` call is used to bind the socket with a port no.
eg. - `bind (socket, port_address)`
- `listen()` call is to enable the port to wait for any incoming requests. This call is done by the server to provide the limit of the connection requests allowed with the server.
eg. `listen (socket, size)`
Here size is used to pass request limit

Aney Thakur B-50

Terna Engineering College, Navi Mumbai

- accept() call is used to issue a request to the accept() function to accept incoming connections

- Syntax:

accept (new_socket, socket)

- If the accept() call is successful then a new socket is returned for future connection with the respective client

Stages - for client side programming

- Creating a socket using the socket() call function

- Syntax -

use socket;

- Connect() call is used to connect the socket with the server by using specific address
connect (socket, address)

Here, address is similar to in bind() except that it contain the IP address of remote server

Amey Thakur B-50

Terna Engineering College, Navi Mumbai

B) Perl database connectivity.

- Creating database programs is one of the most common use of perl.
- Using perl, we can create robust web applications along with a database to manage all data.
- It provides excellent support for interfacing and a broad range of data formats.
- For connecting to and querying a database, perl provides a module called DBI.
- DBI is a database interface for communicating with database servers that use Structured query language (SQL).
- Accessing a database in perl generally takes 2 steps. The DBI module provides an API. For database access a program uses the function of DBI to manipulate the database.
- The second stage of database access from perl is a database driver (DBD) module.
- Each different database system requires its own drivers. This approach allows a perl database application program to be relatively independent of the particular database it will access.

Terna Engineering College, Navi Mumbai

Database Independent Interface (DBI)

- It provides an independent interface for perl programs
- The perl code does not depend on the database running in the back end
- DBI module provides abstraction, i.e. we can write our own code without worrying about the database that runs in the back end
- To import the function of the DBI module, we need to import the module with help of use program
- The use DBI program allows us to use DBI module to manipulate the database that we are connecting to.
- Syntax : use DBI;
- connect() method is used to connect to specific database

It takes 3 arguments

- ① A string of 3 values separated by , .
First value specifies that we are using DBI,
Second value specifies database engine
Third value specifies name of the database that you want to connect to.
- ② The username
- ③ The password of local system

Syntax :

```
my $dbh = DBI->connect ("DBI:mysql:test","root","password");
die "can't connect:" . DBI->errstr();
```

Terna Engineering College, Navi Mumbai

The or die statement terminates the program if was unable to establish connection with database with an error message.

- The error() method returns a string that contains error encountered when connecting to the database.
- Prepare() method takes in parameter, the SQL querying to be executed, which is in the form of a string that contains the SQL statements.

Syntax:

```
my $sth = $dbh -> prepare ("Create table emp(id int primary key, name varchar(10), salary int);");
```

- Query is prepared for execution in the above query we created a table with id, name and salary column.

- execute() method executes the query written in the prepared() method. It does not take any argument. It is called using statement handle object created when the prepare statement is executed.

Syntax:

```
$sth -> execute();
```

Terna Engineering College, Navi Mumbai

- `fetchrow()` method is used to retrieve the next row of data from the result of the executed query.
- If a `select` query is executed, then `fetchrow()` method fetches the next row from the result.
- It returns one row from the result which can be assigned to variables. When used in a while loop, we can fetch and display all the rows in the database using `fetchrow()` method.

Syntax:

`($id, $name, $salary) = $sth -> fetchrow();`

- The values of each column are stored in the variable.

Disconnecting:

- Once all the queries are executed we need to disconnect the connection.
This is done by the use of `disconnect()` function
- Syntax:
`$dbh -> disconnect();`

Terna Engineering College, Navi Mumbai

Q6. Write program for file operation

* Opening a file in Read-Only mode and closing it.

```
open ( r, "<", "Hello.txt");
print (<r>);
close (r);
```

* Opening a file in write only mode

```
open ( r, "<", "xyz.txt");
print ("Existing content of xyz.txt". <r>);

open ( w, ">", "xyz.txt");
seek r, 0, 0;
print "In writing to file . . .";
# Writing to xyz.txt using print
print w "no content or this file is changed";
close (w);
seek r, 0, 0;
close (r);
```

* Append mode:

```
open ( r, "<", "abc.txt");
print ("Existing content of abc.txt : " <r> );
# Opening file in append mode.
open ( A, ">>", "abc.txt");
seek r, 0, 0;
print "Appending file . . .";
# Appending to abc.txt using print
print A "Hello!";
close (A);
seek r, 0, 0;
close (r);
```