Modules and Packages

Basic Theory:

A **module** allows you to logically organize your Python code. Grouping related code into a module makes the code easier to understand and use. A module is a Python object with arbitrarily named attributes that you can bind and reference.
Simply, a module is a file consisting of Python code. A module can define functions, classes and variables. A module can also include runnable code.
The import has the following syntax –

```
import module1[, module2[,... moduleN]
```

The from...import * Statement
It is also possible to import all names from a module into the current namespace by using the following import statement –

```
from modname import *
```

Packages in Python
A package is a hierarchical file directory structure that defines a single Python application environment that consists of modules and subpackages and sub-subpackages, and so on.

Examples:

1) Write a program which returns the year and name of weekday

```
import datetime
x = datetime.datetime.now()
print(x.year)
print(x.strftime("%A"))
```

2) Print all the modules, variables and functions that are defined under module random.

```
>>>dir(random)
```

3) Import datetime module and print current date and time.

```
import datetime
x = datetime.datetime.now()
print(x)
```

Import Calendar module and print the calendar of any year.

```
import calendar
print ("The calender of year 2012 is : ")
print (calendar.calendar(2012,2,1,6))
```

4) Print the name of days in a week.

```
import calendar
for day in calendar.day_name:
    print(day)
```

Assignments:

1) . Check whether a given year is leap year or not. Provide a range of years and display how many leap years are there.

2) Write a program which will print the month name and first Monday in a given year.

```
import calendar
```

3) . Print the current date and time in the following format.

```
year: 2018
month: 12
day: 24
time: 04:59:31
date and time: 12/24/2018, 04:59:31
```

4) Write a Python program importing random which outputs the frequency of occurrence of each faces for rolling a dice 6000 times.

Package Example:

1) Create a package which contains five modules fish, birds, amphibians, mammals, and reptiles. Each module contains two functions example and characteristics.

2) Create a module which contains the functions printing several patterns. Each function will take input the number of rows and/or columns as required.

*	*	*	1	1
* *	* *	* *	1 2	2 3
* * *	* * *	* * *	1 2 3	4 5 6
* * * *	* * * *	* * * *	1 2 3 4	7 8 9 10
* * * * *	* * * * *	* * * * *	1 2 3 4 5	11 12 13 14 15