Time: maximum 3 hrs.

**Python Questions**

1. What is the statement that can be used in Python if a statement is required syntactically but the program requires no action?
2. If you are asked to find 3 rd power of a number without using python operator, what will be the alternative way you prefer?
3. If a binary value is 1100. How can you make this to 110000 using python?
4. I have a list of 25000 items. How will you identify an item is present in that list?
5. Is it possible to make infinite loop in python? If yes how?
6. We know that a for/while loop continues till the end of a particular sequence/series. How can you stop an iteration before it ends?
7. Difference between map, filter, and reduce?
8. How can you call different functions simultaneously in python? Is there any in-built module helpful for that?
9. How to check the type of a data in python?
10. Why python called dynamically typed?

**Django Questions**

1. Is django follows model-view-controller pattern?
2. Can you pass python variable to html page?
3. Django provides admin panel. If I wish to see only two tables in there, what should I do?
4. What is Django ORM?
5. What is the use of session?
6. What are static files in Django?
7. In Django what is query for view all item from model
8. Explain Django models
9. Explain Django Views
10. Explain Django urls

**Python coding Questions**

1. Write a program to create a list. Reverse the list without using any function.
2. Write a program to receive a sequence of comma-separated 10 letters as string from user and generate a list and a tuple which contains all letters. Take 2 nd index item from the list to a variable by removing that item from the list. Add this removed element to 5 th index without deleting any other items. Replace 8 th to 10 th elements with new letters.
3. Print the following pattern



**Django Coding Questions**

1. Create a project and application in Django. There will be three tables like below.

class Meetings(models.Model):  
    Meetingid = models.IntegerField()  
    Meeting\_Date= models.DateField()  
    Start\_time = models.TimeField()  
    End\_time = models.TimeField()  
    Createdby = models.CharField(max\_length=50)  
    Createdbytime = models.TimeField()  
  
class Participant(models.Model):  
    Participant\_meet = models.ForeignKey(Meetings, on\_delete=models.SET\_NULL, null=True)  
    First\_name = models.CharField(max\_length=50)  
  
class Details(models.Model):  
    Details\_meet = models.ForeignKey(Meetings, on\_delete=models.SET\_NULL, null=True)  
    Decision = models.TextField()  
    First\_name = models.CharField(max\_length=50)  
    Date = models.DateField(auto\_now\_add = True)

Superuser must be created. Home page will be login page. Only superuser can login.

After login from home page it should reach index page.

Index page can only be accessed by logged in member.

Index page must have two links – logout, new meeting. ‘New meeting’ link should navigate to ‘Add member’ page where admin can add a new member which must be stored in ‘Meetings’ table.

After submitting from  ‘Add member’ page, it should navigate to ‘Add participants’ page.

Here new participants can be added. Participant details should be stored in ‘Participant’ table.

After participants are added, it should navigate to “Member’s decision” page.

Decision of each member must be added and should be stored in ‘details’ page.

Member should be taken from a selection list and they should be from that particular meeting.

One member can have multiple decisions. This page should navigate to ‘Submit details’ page.

This page should display meeting id, meeting date, start time, end time, meeting created time (same as logged in time), participants of meeting.

This page should navigate to a page where meeting can be searched by date (yyyy-mm-dd) . It should display ‘start time’ and ‘end time’ of meetings of that particular date.

Each row must have a ‘view’ button. ‘view’ button can only work on that particular time interval of meeting.

After clicking ‘view’ button details of that particular meeting should be displayed on next page. Details include- date of meeting, start time, end time, participants.

Also decisions should be displayed in a table that have column headings – serial number, decision, participant.