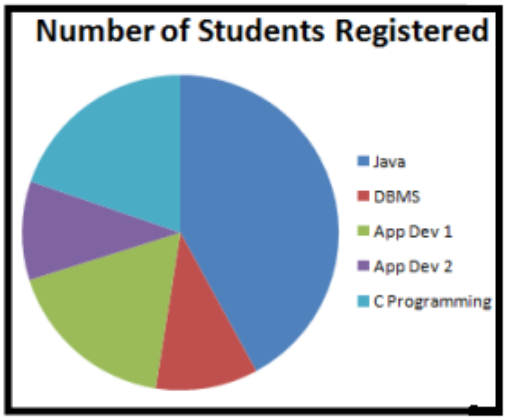
BSCCS2003: Graded Questions with Solutions
Week 3

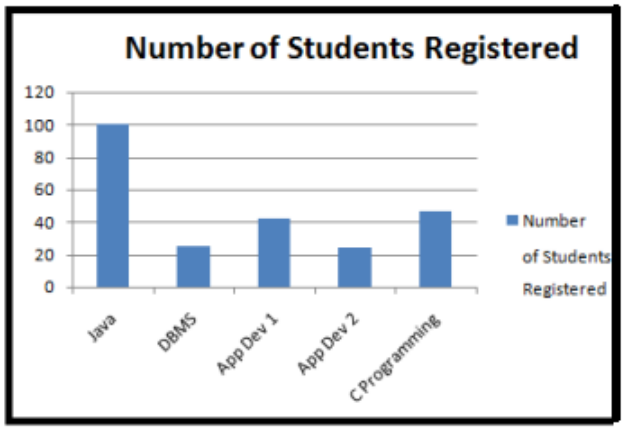
1. Consider the images given below and identify the correct statement(s). [MSQ : 1 point]

Course Name	Number of Students Registered
Java	100
DBMS	25
App Dev 1	42
App Dev 2	24
C Programming	47

A



B



√ Model is an object which represents data in the table.

√ Chart A and chart B are two different views of the same data table.

√ Controller is an intermediate between the model and the view.

◯ None of the above

Solution:
In MVC design pattern, a model is an application object which can store the application data. The View represents data in specified ways. Controller is the one that manipulates the model and updates the view.

2. Consider the pyhtml code given below and identify the resulting HTML code. [MCQ : 2 points]

```
from pyhtml import *

t = html(
    head(
        title('Sample Pyhtml File')
```

```
        ),
    body(
        header(img(src='./logo.png'))
        )
        )
print (t.render())
```

√
```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Sample Pyhtml File
    </title>
  </head>
  <body>
    <header>
      <img src="./logo.png"/>
    </header>
  </body>
</html>
```

◯
```
<!DOCTYPE html>
<html>
  <head>
    <title>
      Awesome website
    </title>
    <script src="http://path.to/script.js"></script>
    <title>
      Sample Pyhtml File
    </title>
  </head>
  <body>
    <header>
      <img src="./logo.png"/>
    </header>
  </body>
</html>
```

◯
```
  <head>
    <title>
    </title>
  </head>
  <body>
    <header>
```

```
        <img src="./logo.png"/>
      </header>
    </body>
  </html>
```

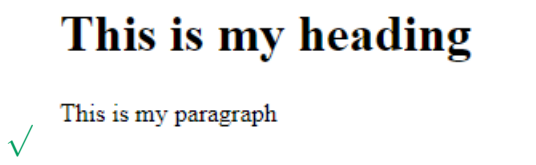
◯ None of the above

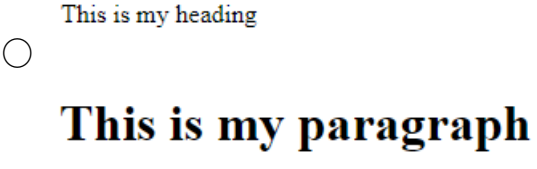
Solution:
Option 1 : The first code snippet will be the output of given template code.
Option 2 : An additional script tag is included which is not defined in the template code.
Option 3: The title is missing in the output code.

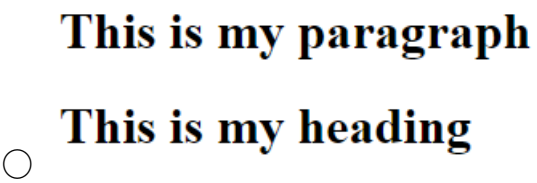
3. What will be the output of the following Python code if rendered using a browser?

```
def Generate(t1):
    begin = "<!DOCTYPE html>\n<html>\n<body>\n"+t1+"\n</body>\n
    </html>"
    return begin
def div(head,body):
    return "<div>"+head+"\n"+body+ "\n</div>"
def H1(content):
    return "<h1>"+content+"</h1>"
def p(content):
    return "<p>"+content+"</p>"
Out = Generate(div(H1("This is my heading"),
      p("This is my paragraph")))
```

[MCQ: 2 points]

√ 

◯ 

◯ 

This is my paragraph

This is my heading

○

Solution: The given python code will be rendered into the HTML file given below:

```
<!DOCTYPE html>
<html>
<body>
<div><H1>This is my heading</H1>
<p>This is my paragraph</p>
</div>
</body>
</html>
```

The HTML file will display the webpage as given in option 1.

4. Consider the image given below and identify the correct pyhtml code to generate equivalent HTML output. [MCQ : 2 points]

√
```
from pyhtml import *

t = html(
    head(
        title('Sunflower Image')
        ),
    body(
        h1('This is Heading 1'),
        b('This text is in bold'),
        i('This text is in Italics'),
        u('This text is in Underlined'),
        em('This text is Emphasized')
```

```
        )
        )

print (t.render())
```

◯
```
from pyhtml import *

t = html(
    head(
        title('Sunflower Image')
        ),
    body(
        h1('This is Heading 1'),
        b('This text is in bold'),
        i('This text is in Italics'),
        u('This text is in Underlined'),
        em('This text is Emphasized')
        )
        )

print (t.show())
```

◯
```
from pyhtml import *

t = html(
    head(
        title('All tags')
        ),
    body(
        h1('This is Heading 1'),
        b('This text is in bold'),
        i('This text is in Italics'),
        u('This text is in Underlined'),
        em('This text is Emphasized')
        )
        )

print (t.display())
```

◯ None of the above

Solution:
Option 1: The code here is a valid code and will result in correct output without any errors.

Option 2: The code snippet here is similar to option 1 except that it uses t.show() method instead of t.render() which is not a valid pyhtml rendering method.
Option 3: The code snippet here is similar to option 1 except that it uses t.display() method instead of t.render() which is not a valid pyhtml rendering method.

5. What will be the output of the following Python code? [MCQ: 3 points]

```
from jinja2 import Template
my_statement = Template("The special numbers are: {% for n in range(11)
                        if n%2 == 0 %} {{n%3}}" "{% endfor %}")
out = my_statement.render()
print(out)
```

- ◯ The special numbers are: 1 0 1 0 1 0 1 0 1 0 1 0 1 0
- √ The special numbers are: 0 2 1 0 2 1
- ◯ The special numbers are: 0.5 1.0 1.5 2.0 2.5 3.0 3.5 4.0 4.5 5.0 5.5 6.0 6.5 7.0
- ◯ The special numbers are: 0.33 0.67 1.0 1.33 1.67 2.0

Solution: In the Template function, the {% %} is a block used to provide conditional statements and the content to be printed is given inside {{ }}. In Python, the expression n%3 will return the remainder when n is divided by 3.

6. What will be the output of following Python code? [MCQ: 1 point]

```
from jinja2 import Template
statement1 = Template("IIT Madras provides diploma in {{Value_1}} {{Value_2}}")
statement2 = Template("IIT Madras provides degree in {{Value_1}}{{Value_2}}")
out1 = statement1.render(Value_1 = "programming")
out2 = statement2.render(Value_2 = "data science")
print(out1)
print(out2)
```

- √ IIT Madras provides diploma in programming
  IIT Madras provides degree in data science
- ◯ IIT Madras provides diploma in data science
  IIT Madras provides degree in programming
- ◯ IIT Madras provides diploma in programming, data science
  IIT Madras provides degree in programming, data science
- ◯ The code will generate parsing error.

Solution: The Template class of Jinja2 simply ignores those brackets whose actual content is not provided by the render method. It does not throw any error.

7. Which of the following Python codes will generate the HTML code given below? [MCQ : 3 points]

```
<!DOCTYPE html>
<html>
  <head></head>
  <body>
    <h1>
       Heading 1
    </h1>
    <em>
       My name is ram
    </em>
    <strong>
        bold
    </strong>
    <div>
      <h2>
         Heading 2
      </h2>
      <div>
         This is a nested div section
      </div>
    </div>
    <div>
      <h3>
          Heading 3
      </h3>
      <p>
          This is a paragraph
      </p>
    </div>
  </body>
</html>
```

√
```
import pyhtml as ht
t=ht.html(ht.head(),
ht.body(
   ht.h1('Heading 1'),ht.em('My name is ram'),ht.strong("bold"),
   ht.div(ht.h2("Heading 2"),
   ht.div("This is a nested div section")),
   ht.div(ht.h3('Heading 3'),
   ht.p("This is a paragraph")),
```

```
))
print(t.render())
```

◯
```
import pyhtml as ht
t=ht.html(ht.head(),
ht.body(
   ht.h1(‘Heading 1’) ht.em(‘My name is ram’) ht.strong("bold")),
   ht.div(ht.h2("Heading 2"),
   ht.div("This is a nested div section")),
   ht.div(ht.h3(’Heading 3’),
   ht.p("This is a paragraph")),
)
print(t.render())
```

◯
```
import pyhtml as ht
t=ht.html(ht.head()
ht.body(
   ht.h1(‘Heading 1’),ht.em(‘My name is ram’),ht.strong("This is bold"),
   ht.div(ht.h2("Heading 2"),
   ht.div("This is a nested div section")),
   ht.div(ht.h3(’Heading 3’),
   ht.p("This is a paragraph")),
))
print(t.render())
```

◯ None of the above

Solution: This question is based on the Python library called pyhtml to generate a valid HTML file using Python. (Reference: Timeframe 3:40 in Lecture 3.6, Tools-part II)