



Complete, fold and seal
(see note 2 below)

FORENAME(S): MIRCEA-RADU-SERBAN

12 Pages

1. Please give the information requested in the boxes below.
2. Enter your name in the space provided to the right, fold over and seal down (using the label provided) to hide your name.
3. The answers must be legibly written. The Examiners can either ignore illegible script or have it typed and the expense of typing charged to the candidate.
4. Write your answers in ink.
5. Begin each answer on a fresh page. Do all rough work in the answer book and cross it through.
6. No part of this book is to be torn out; it must be handed in complete.
7. If more books than one are used, fasten them together using the tags available from the invigilator(s).
8. **Every book must be handed in. In no circumstances may answer books, used or unused, be taken out of the examination venue.**

[illegible]

492
75
66%
AT 1/6/18

1)

- a) Typescript first of all enables an OOP approach through the use of classes. As can be seen in the Position class, Typescript provides support for classes with attributes, properties or accessor methods, constructor as well as type declaration. Next, as can also be seen in the Position class, various levels of protection towards its members (in this example attributes, marked with "private") are supported. Finally classes may also be abstract as seen in the Shape class. ← why abstract?

On the other hand, interfaces are also supported as can be seen with the IDrawable interface, which in turn is implemented in the Shape abstract class. In this case, Shape implements the IDrawable interface.

Finally, inheritance is also supported as can be seen in the Rectangle class which inherits from Shape with the "Extends" notation.

It has to be noted however that the Shape class is abstract, meaning that any class that is derived from it ~~must not override~~ has to respect some implementation details, such as types and method functionality. As such it can be said that an abstract class works as an extended interface towards derived classes.

- b) First of all, the Position class lacks set methods.
e.g.

```
set X(): number
{
  this.x = value;
}
```

```
set Y(): number
{
  this.y = value;
}
```

Secondly, the Shape class also lacks a set method.

```
set El: JAwng.
```

```
{
  return this.el = value
}
```

Next, the Rectangle class has an extra character in its position type, "Position Z", which should be removed.

The correct notation is position: Position. Next, in the constructor, the named JAwng element param is not assigned. Finally, the "html" should have two minutes.

Finally, the "html" to the draw() method should be replaced by innerHTML for correction.

Code for a picture class may be as follows:

```
class Picture
```

```
{ private el: JAwng.
```

```
  - position: Position
```

```
  - itemLink: Link < JDrawable >.
```

```
constructor (el: JAwng, position: Position, link: Link < JDrawable >)
```

```
{ this.el = el;
  this.position = position;
  this.itemLink = link; }
```

```
drawItems():
```

```
{ for each (item as Link - item Link) {
```

```
  item.draw();
}
```

2.5

1) XML.

```

<employee>
  <name> Fred Bloggs </name>
  <department> Engineering </department>
  <roles>
    <role> Designer </role>
    <role> Programmer </role>
    <role> Tester </role>
  </roles>
  <email>
  </email>
  <telephone>
  </telephone>
</employee>
</employee>
</roles>
<email> fred.bloggs@company.com </email>
<telephone> 01523847983 </telephone>
</employee>

```

JSON:

```

{
  "name": "Fred Bloggs",
  "department": "Engineering",
  "roles": ["Designer", "Programmer", "Tester"],
  "email": "fred.bloggs@company.com",
  "telephone": "01523847983"
}

```

8

~~to return an object of arrays.~~

2)

a) Explanation.

REST stands for "Representational State Transfer". An such, REST is a means of transferring data formatted as either JSON or XML using only the HTTP transfer methods.

GET - fetch data
 POST - send data
 PUT - update data
 DELETE - delete data.

Using these methods the webpage talks with ^{data} server
 the server which in turn talks to the server.
 The server in turn knows what to do depending on the
 method used eg.

GET Students would retrieve ~~students data~~ a list of
 student under either \$500 or XMC which is afterwards
 passed by the callback code of the request ~~output~~
 into either an object or ~~an~~ rather type of data.
 fit for the web application.

These actions are usually done asynchronously
 using AJAX (Asynchronous JavaScript and XML)
 for dynamic fetching of content (no page reloading).

Benefits

AJAX - AJAX provides ~~dynamic~~ content to the user dynamically
 e.g. viewing different datasets without having to reload
 the webpage. It may also provide frequent ~~real time~~ updates
 in an accurate manner, adding to a more intuitive feel.

REST - An HTTP method ^{is} ~~implemented~~ a great number of
 languages ^{that support} a REST approach ensures a language and/or
 platform agnostic means towards accessing ~~and~~ sending
 data.

- b) The first design can't work as web application can't talk
 directly to server. They must have ~~some~~ Web app and
 server need to establish a proxy port through which they
 communicate.

Answer continues two pages forward.

3/

a) The 4 V's of Big data are as follows:

Volume - deals with the fact that data is being stored and processed in very large quantities.

Variety - deals with the fact that data can come under different formats (CSV, JSON, XML etc.).

Variability - deals with the fact that data may be corrupt or inconsistent.

Velocity - deals with the speed at which data is being processed, ~~to~~ received etc.

b) ^{The} Batch Layer's purpose is to do a large number of operations on the ^{incoming} original data ^{set} (Master Dataset) because the terminology "Batch" as actions are executed in "batch" ~~of~~ one after the other consecutively.

The Ingestion Layer afterwards which is lower granular data into multiple streams according to multiple variables or according to user request, for easy access.

Finally, the Query Layer is also an option and provides the user with processed data as it comes into the data lake. It may present itself under multiple views ~~and~~ depending on the user's requirements.

The incoming data may arrive from various sources such as smartphones, IoT devices and all other sort of devices with networking capabilities.

Next, the query specifies what type of data to get as well

on the rate at which it is received etc.

- 1) ^{Layer} Speed Error - ~~The data may be not be in time with~~
if the data is not accurate or is too old, a new.
real time scene is generated instead.

Batch Layer - The batch layer may reprocess its batch.
operations or attempt them on other data. Corrupted ^{for} ~~data~~ ^{immediate}
data is not removed so to ensure immutability.

Error Layer - The error layer may try to reprocess.
attempts with the batch layer for reprocessing, again.
any type of data is not discarded.

Eventual consistency may ~~may~~ be guaranteed represents
the characteristic of data to be consistent enough.
and/or useful enough after processing or reprocessing.
it also includes the characteristic of being as up-to-date
as required.

- 2) Their relation works as ~~the same as~~ the server works as
a proxy to the service.

Additionally, a web application can talk to a server,
directly. As there is no mechanism for checking
transactions etc. There is ~~nothing~~ no mechanism for
doing REST transactions.

A possible solution may be as follows.

