Name: K.V.R saicharan
Roll no: 320227720012
1st semester
Sub: OOPS with Java programming

1)Defne Servlet and explain the servlet life cyccle?
a)servlet:: A servlet is a Java programming language class that is used to
extend the capabilites of servers that host applicatons accessed byc means of a request-response programming model. Although servlets can respond to anyc
tycpe of request, theyc are commonlyc used to extend the applicatons hosted byc
web servers. For such applicatons, Java Servlet technologyc defnes HTTPspecifc
servlet classes.
servlet lifecycle::
The life cyccle of a servlet is controlled byc the container in which the servlet has
been deployced. When a request is mapped to a servlet, the container performs
the following steps.
- Loading a Servlet.

- Initalizing the Servlet.
- Request handling.
- Destroycing the Servlet.

a) Loading a Servlet: The frst stage of the Servlet lifecyccle involves loading and initalizing the Servlet byc the Servlet container. The Web container or Servlet Container can load the Servlet at either of the following two stages :

- Initalizing the context, on confguring the Servlet with a zero or positive integer value.
- If the Servlet is not preceding stage, it mayc delayc the loading process until the Web container determines that this Servlet is needed to service a request.

b) Initalizing the Servlet: Afer the Servlet is instantated successfullyc, the Servlet container initalizes the instantated Servlet object. The container initalizes the Servlet object byc invoking the Servlet.init(ServletConfg) method which accepts ServletConfg object

reference as parameter.
The Servlet container invokes the Servlet.init(ServletConfig) method only once, immediately after the Servlet.init(ServletConfig) object is instantiated successfully. This method is used to initialize the resources, such as JDBC datasource.
c) Handling Request: After initialization, the Servlet instance is ready to serve the client requests. The Servlet container performs the following operations when the Servlet instance is located to service a request :
- It creates the ServletRequest and ServletResponse objects. In this case, if this is a HTTP request, then the Web container creates HttpServletRequest and HttpServletResponse objects which are subtypes of the ServletRequest and ServletResponse objects respectively.
- After creating the request and response objects it invokes the Servlet.service(ServletRequest, ServletResponse)

method by passing the
request and response objects.
d) Destroying a Servlet: When a Servlet
container decides to destroy the
Servlet, it performs the following operations,
- It allows all the threads currently running in
the service method of the
Servlet instance to complete their jobs and get
released.
- After currently running threads have
completed their jobs, the Servlet
container calls the destroy() method on the
Servlet instance.

There are three life cycle methods of a Servle
t :
- init()
- service()
- destroy()
- init(): The Servlet.init() method is called by the
Servlet container to
indicate that this Servlet instance is instantated
successfully and is about
to put into service.
- service(): The service() method of the Servlet is
invoked to inform the

Servlet about the client requests.
- This method uses ServletRequest object to collect the data requested by the client.
- This method uses ServletResponse object to generate the output content.
- destroy(): The destroy() method runs only once during the lifetme of a Servlet and signals the end of the Servlet instance

2) Explain about jdbc driver?
a)JDBC Driver is a sofware component that enables java applicaton to interact with the database. There are 4 types of jdbc drivers.
1. jdbc odbc bridge driver
2. Natve-API driver (partally java driver)
3. Network Protocol driver (fully java driver)
4. Thin driver (fully java driver)

1) jdbc bridge driver.
The jdbc bridge driver uses ODBC driver to

connect to the database. The JDBC-driver converts JDBC method calls into the ODBC function calls. This is now discouraged driver.

Advantages:

- easyc to use.
- can be easilyc connected to anyc database.

Disadvantages:

- Performance degraded because JDBC method call is converted into the ODBC functon calls.
- The jdbc driver needs to be installed on the client machine.

2) Natve-API driver (partallyc java driver):

The Natve API driver uses the clientside libraries of the database. The driver converts jdbc method calls into natve calls of the database API. It is not writen entrelyc in java.

Advantage:

- performance upgraded than jdbc odbc bridge driver.

Disadvantage:

- The Natve driver needs to be installed on the each client machine.
- The Vendor client libraryc needs to be installed

on client machine.
3.) Network Protocol driver (fully java driver): The Network Protocol driver uses middleware (applicaton server) that converts JDBC calls directly or indirectly into the vendor-specific database protocol. It is fully writen in java.
Advantage:
o No client side library is required because of applicaton server that can perform many tasks like auditng, load balancing, logging etc.
Disadvantages:
o Network support is required on client machine.
o Requires database-specific coding to be done in the middle ter.
o Maintenance of Network Protocol driver becomes costly because it requires database-specific coding to be done in the middle ter.
4.) Thin driver (fully java driver): The thin driver converts jdbc calls directly into the vendor-specific database protocol. That is why it is known as thin driver. It is fully writen in Java language

Advantage:
o Beter performance than all other drivers.
o No sofware is required at client side or server side.
Disadvantage: Drivers depend on the Database.

3) Explain Java fxui Controls?
a) The graphical user interface of everyc desktop applicaton mainlyc considers UI elements, laycouts and behaviour.
The UI elements are the one which are actuallyc shown to the user for interacton or informaton exchange. Laycout defnes the organizaton of the UI elements on the screen. Behaviour is the reacton of the UI element when some event is occurred on it.
However, the package javafx.scene.control provides all the necessaryc classes for the UI components like Buton, Label, etc. Everyc class represents a specifc UI control and defnes some methods for their stycling.

*Label- Label is a component that is used to defne a simple text on the screen.
Tycpicallyc, a label is placed with the node, it describes.
*Buton- Buton is a component that controls the functon of the applicaton.
Buton class is used to create a labelled buton.
*RadioButon-The Radio Buton is used to provide various optons to the user.
The user can onlyc choose one opton among all. A radio buton is either
selected or deselected.
*Checkbox- Check Box is used to get the kind of informaton from the user
which contains various choices. User marked the checkbox either on (true) or
off(false).
*TextField- Text Field is basicallyc used to get the input from the user in the
form of text. javafx.scene.control.TextField represents TextField.
* PasswordField-PasswordField is used to get the user's password. Whatever is
tycped in the passwordfeld is not shown on the

screen to anycone.
*Hycperlink- HycperLink are used to refer anyc
of the webpage through ycour
appicaton. It is represented byc the class
javafx.scene.control.HyperLink
*Slider- Slider is used to provide a pane of optons
to the user in a graphical
form where the user needs to move a slider over
the range of values to select
one of them.