1. https://fts.capgemini.com/private/6041246513748/02-Styling-With-Bootstrap.mp4 link valid until 2019-07-24 10:00 UTC (SSO protected)

2. https://fts.capgemini.com/private/6041246513748/03-Databinding.mp4 link valid until 2019-07-24 10:00 UTC (SSO protected)

3. https://fts.capgemini.com/private/6041246513748/01-Creating-Angular-Component.mp4 link valid until 2019-07-24 10:00 UTC (SSO protected)

4. https://fts.capgemini.com/private/26002741398718/05-Structural-Directives.mp4 link valid until 2019-07-24 10:00 UTC (SSO protected)

5. https://fts.capgemini.com/private/26002741398718/06-Attribute-Directives.mp4 link valid until 2019-07-24 10:00 UTC (SSO protected)

6. https://fts.capgemini.com/private/26002741398718/07-Custom-Directives.mp4 link valid until 2019-07-24 10:00 UTC (SSO protected)

7. https://fts.capgemini.com/private/26002741398718/04-Nesting-Components-Component-Communication.mp4 link valid until 2019-07-24 10:00 UTC (SSO protected)

8. https://fts.capgemini.com/private/82264284286251/08-Angular-Pipes.mp4 link valid until 2019-07-24 12:00 UTC (SSO protected)

9. https://fts.capgemini.com/private/65549781685073/16-HTTP-Interceptors.mp4 link valid until 2019-07-25 11:00 UTC (SSO protected)

10. https://fts.capgemini.com/private/65549781685073/13-Handling-Errors.mp4 link valid until 2019-07-25 11:00 UTC (SSO protected)

11. https://fts.capgemini.com/private/65549781685073/14-Implementing-CRUD-Operations-In-Service.mp4 link valid until 2019-07-25 11:00 UTC (SSO protected)

12. https://fts.capgemini.com/private/79310384560024/11-Working-With-HttpClient.mp4 link valid until 2019-07-25 11:00 UTC (SSO protected)

13. https://fts.capgemini.com/private/79310384560024/12-Rendering-Service-Data-In-View.mp4 link valid until 2019-07-25 11:00 UTC (SSO protected)

14. https://fts.capgemini.com/private/65549781685073/15-Creating-View-To-Perform-CRUD.mp4 link valid until 2019-07-25 11:00 UTC (SSO protected)

15. https://fts.capgemini.com/private/79310384560024/09-Angular-Services.mp4 link valid until 2019-07-25 11:00 UTC (SSO protected)

16. https://fts.capgemini.com/private/79310384560024/10-Creating-Custom-Services.mp4 link valid until 2019-07-25 11:00 UTC (SSO protected)



https://fts.capgemini.com/private/2280600616960/Lifecycle-Hooks-Part01.mp4 link valid until 2019-07-26 9:00 UTC (SSO protected)

https://fts.capgemini.com/private/65295582418368/24-Component-Communication-Observables.mp4 link valid until 2019-07-26 12:00 UTC (SSO protected)

https://fts.capgemini.com/private/65295582418368/Demos.zip link valid until 2019-07-26 12:00 UTC (SSO protected)

https://fts.capgemini.com/private/28419940271047/18-Angular-Routing-Introduction.mp4 link valid until 2019-07-26 9:00 UTC (SSO protected)

https://fts.capgemini.com/private/28419940271047/19-Configuring-Routes.mp4 link valid until 2019-07-26 9:00 UTC (SSO protected)

https://fts.capgemini.com/private/2280600616960/Lifecycle-Hooks-Part02.mp4 link valid until 2019-07-26 9:00 UTC (SSO protected)

https://fts.capgemini.com/private/28419940271047/20-Passing-Data-Guarding-Routes.mp4 link valid until 2019-07-26 9:00 UTC (SSO protected)

https://fts.capgemini.com/private/28419940271047/21-Nested-Routes.mp4 link valid until 2019-07-26 9:00 UTC (SSO protected)







































https://fts.capgemini.com/pubpwd/4633615948962/jdk.7z



username: wxpamhquq

Password: RxWyWP5g8A























agile training link --

https://capgemini.skillport.com/skillportfe/main.action?path=browsecatalog//\_fcpaths\_L1NraWxsU29mdFNQOC9JVFNraWxsczEvU29mdGVEZXNpcG1lbnQvQWdpbGVEZXZlbG9wbWVudC9zZF9hZGV2\_fcpathe\_#browse/85b61bdc-7816-4368-a168-c43c38a925c8





------------------------------------------------

https://fts.capgemini.com/pubpwd/18799960327168/Module-2.7z link valid until 2019-05-14 6:00 UTC (public access with local password)

Additional credentials, to enter in specific popup box when prompted for authentication.

username: aztystvaw

Password: YHPuyfwCgX



-----------------------------------------------------------------



Project interview questions



1. oops concept details with eg

2. ways to achieve abstraction

3. Exceptions type wid eg

4. what is throw throws

5. difference array and arraylist

6. difference linked list and arraylist and examples

7. how to create a hashmap

8. spring mvc flow

9. hibernate and its config file

10. string pool concept

11. what is interface

12. diff between public protected private default

13. marker interface wid detail example

14. diff between inner join and outer join

15. diff left outer and right outer

16. trigger

17. features of java 8

18. lambda expression

19. abstract class--- can we write abstract class without abstract method

20. class and object definition

21. java-permgen-metaspace





--------------------------------------



Module2 sample code



https://fts.capgemini.com/pubpwd/60613204179962/Module2-SampleCode.zip link valid until 2019-05-01 10:00 UTC (public access with local password)

https://fts.capgemini.com/pubpwd/60613204179962/StudentApplication.7z link valid until 2019-05-01 10:00 UTC (public access with local password)

username: decodnoaf

Password: KsWWZX9trA





------------------------------------------------



JEE with cloud training material--



https://fts.capgemini.com/pubpwd/5321565542988/Newfolder.7z



username: repycmpjd

Password: yQ9NTku5hu



---------------------------------------------------































1. Mr Dighikar --- Prasanna , Plot 253, Sector 25, oppsite to cosmos bank/dnyanprbhodini school, pradhikaran, nigdi, pune 09766844149



Looking for one Room Mate for already occupied 3 BHK Flat in Nigdi. Price includes rent, chef, maid, wifi, current bill and groceries for every month. Advance 10000. TV and Washine Machine Available.



Preeti Vijaykumar Matta

Property Address: 430/24, Sector 24, Pradhikaran, Nigdi Near Lokmanya Hospital, Pune - 411044

Nagaraju Tadisetti -- 9028591034





















































































https://www.webopedia.com/quick\_ref/computer-certifications.html



https://o7planning.org/en/10285/create-a-simple-java-web-application-using-servlet-jsp-and-jdbc



https://www.microsoft.com/en-us/learning/browse-all-certifications.aspx







https://www.journaldev.com/2651/spring-mvc-exception-handling-controlleradvice-exceptionhandler-handlerexceptionresolver











































------------------------------------------------------------------------------------------------------------------------------





















The file(s) can be downloaded with the following URL(s) and credentials:

https://fts.capgemini.com/pubpwd/8572574361520/MeghanaEmployeeMaintenanceSystem.zip link valid until 2018-08-27 13:00 UTC (public access with local password)

Additional credentials, to enter in specific popup box when prompted for authentication.

username: njbuwfeyo

Password: JENT8UpdDS











?????????????????????????????????







































\*\*\*\*\*\*\*\*\*\*\*\*Exception in Mongodb\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*



package com.cg.controller;

import java.util.List;

import javax.validation.Valid;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.web.bind.annotation.PathVariable;

import org.springframework.web.bind.annotation.RequestBody;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.bind.annotation.RequestMethod;

import org.springframework.web.bind.annotation.RestController;

import com.cg.entity.Student;

import com.cg.exception.StudentNotFoundException;

import com.cg.service.StudentService;

@RestController

@RequestMapping("/api")

public class StudentController

{

private final Logger LOG = LoggerFactory.getLogger(getClass());

@Autowired

private StudentService studentService;

@RequestMapping(value="/create",method=RequestMethod.POST)

public Student addNewStudent(@Valid @RequestBody Student student)

{

LOG.info("Adding new Students");

return studentService.addNewStudent(student);

}

@RequestMapping(value="/{studentRoll}",method=RequestMethod.DELETE)

public Student deleteStudent(@PathVariable String studentRoll)

{

LOG.info("Deleting Student detail");

Student student= studentService.deleteStudent(studentRoll);

if(student==null)

{

throw new StudentNotFoundException("Student Not Found");

}

return student;

}

@RequestMapping(value="/getall",method=RequestMethod.GET)

public List <Student> getAllStudent()

{

LOG.info("Getting details of students");

List <Student> list =studentService.getAllStudent();

if(list.isEmpty())

{

throw new StudentNotFoundException("Record is empty");

}

return list;

}

@RequestMapping(value="/{studentRoll}",method=RequestMethod.GET)

public Student getStudent(@PathVariable String studentRoll )

{

LOG.info("Getting Student details with Roll: {}",studentRoll);

Student student=studentService.getStudentbyRoll(studentRoll);

if(student==null)

{

throw new StudentNotFoundException("Student Not Found");

}

return student;

}

}

------------------------------------------------------------------------------



























//Merged Mini project



https://fts.capgemini.com/pubpwd/277828304208/UpdatedNewMINIPROJECT.zip link valid until 2018-08-23 8:00 UTC (public access with local password)

username: bstjcraax

Password: E9Q2YroDut





-----------------------------------------------------------------

The file(s) can be downloaded with the following URL(s) and credentials:

https://fts.capgemini.com/pubpwd/46451363875/studentapp.zip link valid until 2018-08-23 8:00 UTC (public access with local password)

Additional credentials, to enter in specific popup box when prompted for authentication.

username: ahfxwfwxg

Password: fFHmFmKWvM



------------------------------------------------------------------













1. Which of the following is not a NoSQL database ?

a) SQL Server

b) MongoDB----ans

c) Cassandra

d) None of the mentioned



2.\_\_\_\_\_\_\_\_ stores are used to store information about networks, such as social connections.

a) Key-value

b) Wide-column

c) Document

d) Graph ---- ans

















1.1

use CgProductDb

1.2

db.createCollection("ProductCollection")

1.3

db.ProductCollection.insert({ \_id:1,item:"Laptop",prodCat :"Electronics",price:30000.0,quantity: 2,orderInfo:{\_id:"O001",orderdate: ISODate("2014-01-01"),address:{street:"66,Airoli",city:"Mumbai",state:"MS",zipcode:700987,coords: [ -73.856077, 40.848447 ]},email:"capgemini@capgemini.com",mobiles:[8888108810]}})

db.ProductCollection.insert({ \_id:2,item:"Mobile",prodCat :"Electronics",price:15000.0,quantity: 10,orderInfo:{"\_id":"O002","orderdate" : ISODate("2010-03-12"),address:{street:"FC Road",city:"Pune",state:"MS",zipcode:40081,coords: [ -23.80007, 30.1123456 ]},email:"icres@ibm.com",mobiles:[9856342189]}})

db.ProductCollection.insert({ \_id:3,item:"Google PixelMobile",prodCat :"Electronics",price:15000.0,quantity: 10,orderInfo:{"\_id":"O003","orderdate" : ISODate("2010-03-12"),address:{street:"FC Road",city:"Pune",state:"MS",zipcode:40081,coords: [ -23.80007, 30.1123456 ]},email:"icres@ibm.com",mobiles:[9856342189]}})

db.ProductCollection.insert({ \_id:4,item:"IPhoneX",prodCat :"Electronics",price:105000.0,quantity: 10,orderInfo:{"\_id":"O004","orderdate" : ISODate("2010-03-12"),address:{street:"FC Road",city:"Pune",state:"MS",zipcode:40081,coords: [ -23.80007, 30.1123456 ]},email:"icres@ibm.com",mobiles:[9856342189]}})

db.ProductCollection.insert({ \_id:5,item:"TV",prodCat :"Electronics",price:24000.0,quantity: 10,orderInfo:{"\_id":"O003","orderdate" : ISODate("2012-06-17"),address:{street:"GT Road",city:"Sahibabad",state:"UP",zipcode:567777,coords: [ -11.80007, 13.1123456 ]},email:"techm@techm.com",mobiles:[7865452222]}})

db.ProductCollection.insert({ \_id:6,item:"Bangles",prodCat :"Jewellery",price:4000.0,quantity: 1,orderInfo:{"\_id":"O004","orderdate" : ISODate("2010-05-16"),address:{street:"Salt Lake",city:"Kolkata",state:"West Bengol",zipcode:222224,coords: [ -67.850007, 9.456666 ]},email:"vaishali@gmail.com",mobiles:[8888108850,9402312123]}})

db.ProductCollection.insert({ \_id:7,item:"Redmi A1",prodCat :"Electronics",price:15000.0,quantity: 10,orderInfo:{"\_id":"O007","orderdate" : ISODate("2010-03-12"),address:{street:"FC Road",city:"Gwalior",state:"MP",zipcode:474011,coords: [ -23.80007, 30.1123456 ]},email:"icres@ibm.com",mobiles:[9856342189]}})

db.ProductCollection.insert([{ \_id:8,item:"Redmi Note 5 Pro",prodCat :"Electronics",price:15000.0,quantity: 10,orderInfo:{"\_id":"O008","orderdate" : ISODate("2010-03-12"),address:{street:"FC Road",city:"Gwalior",state:"MP",zipcode:474011,coords: [ -23.80007, 30.1123456 ]},email:"icres@ibm.com",mobiles:[9856342189]}}])

1.4

db.ProductCollection.find({\_id:1}).pretty()

1.5

db.ProductCollection.find({},{"orderInfo.address.coords":0,"orderInfo.email":0}).pretty()

1.6

db.ProductCollection.update({\_id:3},{$set:{"orderInfo.orderstatus":"pending"}})

1.7

db.ProductCollection.update({item:"Laptop"},{$set:{price:5000}})

1.8

db.ProductCollection.update({item:"Laptop"},{$set:{price:5000}},{multi:true})

1.9

db.ProductCollection.update({\_id:2},{$push:{"orderInfo.mobiles":7509406124}})

1.10

db.ProductCollection.update({item:"CD"},{$set:{prodCat:"Electronics",price:40}})

1.11

db.ProductCollection.updateMany({},{$rename:{item:"productName"}})

1.12

db.ProductCollection.remove({\_id:4})

1.13

db.ProductCollection.remove({productName:{$regex: /^C/}})

1.14

db.ProductCollection.find({$or:[{prodCat:"Electronics"},{productName:"TV"}]}).pretty()

1.15

db.ProductCollection.find({price:{$gt:40000}}).pretty()

1.16

db.ProductCollection.find({price:{$gt:40000}},{productName:1,prodCat:1,price:1}).pretty()

1.17

db.ProductCollection.find({price:{$gt:40000}},{productName:1,prodCat:1,price:1}).limit(3).pretty()

1.18

db.ProductCollection.find({"orderInfo.address.state":"MS"}).pretty()

1.19

db.ProductCollection.find({}).sort({price:-1}).pretty()

1.20

db.ProductCollection.find({}).sort({prodCat:1}).pretty()

1.21

db.ProductCollection.find({}).limit(3).pretty()

1.22

db.ProductCollection.find({}).forEach(function(doc){print("\nProduct Name: "+doc.productName+"\nProduct Category:"+doc.prodCat+"\nOrder Date: "+doc.orderInfo.orderdate+"\nOrder State: "+doc.orderInfo.address.state)})

1.23

db.ProductCollection.find({$or:[{"orderInfo.address.state":"MS"},{"orderInfo.address.state":"UP"}]}).pretty()

1.24

db.ProductCollection.find({$and:[{price :{$gt:40000}},{"orderInfo.address.state":"MS"}]}).pretty()

1.25

db.ProductCollection.find({prodCat:"Electronics", "orderInfo.address.city":"Pune"}).pretty()

1.26

var myCursor = db.ProductCollection.find().sort({prodCat:1})

while(myCursor.hasNext()) { print(tojson(myCursor.next()))}

lab2

2.1- Display the product with “electronic” category.

> db.ProductCollection.aggregate({$match:{prodCat:"Electronics"}}).pretty();

> db.ProductCollection.aggregate([{$match:{prodCat:"Electronics"}}]).pretty()

{

"\_id" : 1,

"prodCat" : "Electronics",

"price" : 40000,

"quantity" : 2,

"orderInfo" : {

"\_id" : "O001",

"orderdate" : ISODate("2014-01-01T00:00:00Z"),

"address" : {

"street" : "66,Airoli",

"city" : "Mumbai",

"state" : "MS",

"zipcode" : 700987,

"coords" : [

-73.856077,

40.848447

]

},

"email" : "capgemini@capgemini.com",

"mobiles" : [

8888108810

]

},

"Product name" : "Laptop"

}

{

"\_id" : 2,

"prodCat" : "Electronics",

"price" : 15000,

"quantity" : 10,

"orderInfo" : {

"\_id" : "O002",

"orderdate" : ISODate("2010-03-12T00:00:00Z"),

"address" : {

"street" : "FC Road",

"city" : "Pune",

"state" : "MS",

"zipcode" : 40081,

"coords" : [

-23.80007,

30.1123456

]

},

"email" : "icres@ibm.com",

"mobiles" : [

8855111111,

[

8855111111

]

],

"mobiles[1]" : 8855111111

},

"orderStatus" : "Pending",

"Product name" : "Mobile"

}



2.2-Group the product category wise and display the count.

> db.ProductCollection.aggregate([{$group:{\_id:"$prodCat",count:{$sum:1}}}]);

{ "\_id" : "Jewellery", "count" : 1 }

{ "\_id" : "Electronics", "count" : 5 }

{ "\_id" : "Clothing", "count" : 1 }

2.3- Display how many number of products available for “Jewellery” category.

> db.ProductCollection.aggregate([{$match:{prodCat:"Jewellery"}},{$group:{\_id:"$

prodCat",count:{$sum:1}}}]);

{ "\_id" : "Jewellery", "count" : 1 }

2.4- Calculate the average amount and average quantity for each category of product.

db.ProductCollection.aggregate([{$group:{\_id:"$prodCat",AverageAmount:{$avg:{$multiply:["$price",$quantity"]}},AverageQuantity:{$avg:"$quantity"}}}]);

{ "\_id" : "Jewellery", "AverageAmount" : 4000, "AverageQuantity" : 1 }

{ "\_id" : "Electronics", "AverageAmount" : 789166.6666666666, "AverageQuantity" : 18.33333333333333

}

{ "\_id" : "Clothing", "AverageAmount" : 102500, "AverageQuantity" : 20.5 }

>

2.5- Find product name, product category, price for those product having quantity greater than 30.

Show only first 3 record.

> db.ProductCollection.find({quantity:{$gt:30}},{productName:1,prodCat:1,price:1

}).pretty().limit(3);



>Db.ProductCollection.aggregate([{$match:{”$quantity”:{$gt:30}}},{$project:{name:1, prodCat:1, price:1}},{$limit:3}])------wrong output

{

"\_id" : 9,

"productName" : "Keyboard",m

"prodCat" : "Electronics",

"price" : 50000

}

{

"\_id" : 10,

"productName" : "Mouse",

"prodCat" : "Electronics",

"price" : 50000

}

{

"\_id" : 15,

"productName" : "Shirts",

"prodCat" : "Clothing",

"price" : 5000

}

2.6-Sort the product quantity wise in ascending order

>db.ProductCollection.aggregate([{$sort:{quantity:1}}]).pretty();

2.7- Display the sum of total sale price category wise.

> db.ProductCollection.aggregate([{$group:{\_id:"$prodCat",TotalSales:{$sum:{$multiply:["$price","$quantity"]}}}}]).pretty();

{ "\_id" : "Jewellery", "TotalSales" : 4000 }

{ "\_id" : "Electronics", "TotalSales" : 4735000 }

{ "\_id" : "Clothing", "TotalSales" : 205000 }

2.8-Find out which product is most selling product.

> db.ProductCollection.aggregate([{$group:{\_id:"$productName",MaxSales:{$max:{$multiply:["$price","$quantity"]}}}},{$sort:{MaxSales:-1}},{$limit:1}]).pretty();

{ "\_id" : "Keyboard", "MaxSales" : 2500000 }

2.9- Find the product list which was ordered in year “2010”

db.ProductCollection.aggregate([{$project:{productName:1, year:{$year: "$orderInfo.orderdate"}}}{$match:{year:2010}}])

> db.ProductCollection.aggregate([{$project:{year:{$year:"$orderInfo.orderdate"}}},{$match:{year:2010}}])

2.10 Display category wise product name and product price.

> db.ProductCollection.aggregate([{$group:{\_id:"$prodCat",ProductDetails:{$push:

{ProductName:"$productName",Price:"$price"}}}}]).pretty();

{

"\_id" : "Jewellery",

"ProductDetails" : [

{

"ProductName" : "Bangles",

"Price" : 4000

}

]

}

{

"\_id" : "Electronics",

"ProductDetails" : [

{

"ProductName" : "Mobile",

"Price" : 15000

},

{

"ProductName" : "TV",

"Price" : 24000

},

{

"ProductName" : "Pen drive",

"Price" : 5000

},

{

"ProductName" : "Laptop",

"Price" : 40000

},

{

"Price" : 40

},

{

"ProductName" : "Keyboard",

"Price" : 50000

},

{

"ProductName" : "Mouse",

"Price" : 50000

}

]

}

{

"\_id" : "Clothing",

"ProductDetails" : [

{

"ProductName" : "Trousers",

"Price" : 5000

},

{

"ProductName" : "Shirts",

"Price" : 5000

}

]

}



----------------------------------------------------------------------------------------------------

//updated new employeee



The file(s) can be downloaded with the following URL(s) and credentials:

https://fts.capgemini.com/pubpwd/40702192503808/EmployeeMaintenanceSystem.7z link valid until 2018-08-20 13:00 UTC (public access with local password)

Additional credentials, to enter in specific popup box when prompted for authentication.

username: uclkydzuu

Password: gBhwR5tK6C



----------------------------------------------------------------------------------------------------

//Mongodb



Author = {

id:1001,

name: "John",

 doj: "12-08-2017",

"book":[

{"name":"Java", "id": 101,"price": 23},

{"name":"Cloud", "id": 102,"price": 45},

{"name":"C", "id": 103,"price": 34}]

};







db.ProductCollection.update({\_id:2},{$push:{"orderInfo.mobiles":[8855111111]}})















































INSERT INTO PERS\_EP\_NSSFDependents (empid, birthdate)

VALUES ('EMP217', CONVERT(DATETIME

, '20/06/2009', 103))









//QueryMapper

package com.cg.emp.dao;

public interface QueryMapper

{

public static final String all="select e.Emp\_ID, e.Emp\_First\_Name, e.Emp\_Last\_Name," +

"e.Emp\_Date\_of\_Birth, e.Emp\_Date\_of\_Joining," +

"e.Emp\_Dept\_ID, d.Dept\_Name, e.Emp\_Grade," +

"e.Emp\_Designation, e.Emp\_Basic, e.Emp\_Gender," +

"e.Emp\_Marital\_Status, e.Emp\_Home\_Address, e.Emp\_Contact\_Num," +

"e.Mgr\_Id from Employee e inner join Department d " +

"on e.Emp\_Dept\_ID=d.Dept\_ID ";

public static final String FindById="select e.Emp\_ID, e.Emp\_First\_Name, e.Emp\_Last\_Name, "

+ "e.Emp\_Date\_of\_Birth, e.Emp\_Date\_of\_Joining, e.Emp\_Dept\_ID, d.Dept\_Name, e.Emp\_Grade, "

+ "e.Emp\_Designation, e.Emp\_Basic, e.Emp\_Gender, e.Emp\_Marital\_Status, "

+ "e.Emp\_Home\_Address, e.Emp\_Contact\_Num, e.Mgr\_Id "

+ "from Employee e inner join Department d "

+ "on e.Emp\_Dept\_ID=d.Dept\_ID "

+ "where e.Emp\_ID = ? ";

public static final String FindByFname="select e.Emp\_ID, e.Emp\_First\_Name, e.Emp\_Last\_Name, "

+ "e.Emp\_Date\_of\_Birth, e.Emp\_Date\_of\_Joining, e.Emp\_Dept\_ID, d.Dept\_Name, e.Emp\_Grade, "

+ "e.Emp\_Designation, e.Emp\_Basic, e.Emp\_Gender, e.Emp\_Marital\_Status,"

+ " e.Emp\_Home\_Address, e.Emp\_Contact\_Num, e.Mgr\_Id "

+ "from Employee e full outer join Department d "

+ "on e.Emp\_Dept\_ID=d.Dept\_ID "

+ "where e.Emp\_First\_Name = ? ";

public static final String FindByLname="select e.Emp\_ID, e.Emp\_First\_Name, e.Emp\_Last\_Name, "

+ "e.Emp\_Date\_of\_Birth, e.Emp\_Date\_of\_Joining, e.Emp\_Dept\_ID, d.Dept\_Name, e.Emp\_Grade, "

+ "e.Emp\_Designation, e.Emp\_Basic, e.Emp\_Gender, e.Emp\_Marital\_Status, "

+ "e.Emp\_Home\_Address, e.Emp\_Contact\_Num, e.Mgr\_Id "

+ "from Employee e inner join Department d "

+ "on e.Emp\_Dept\_ID=d.Dept\_ID "

+ "where e.Emp\_Last\_Name =?";

public static final String FindByDept="select e.Emp\_ID, e.Emp\_First\_Name, e.Emp\_Last\_Name, "

+ "e.Emp\_Date\_of\_Birth, e.Emp\_Date\_of\_Joining, e.Emp\_Dept\_ID, d.Dept\_Name, e.Emp\_Grade, "

+ "e.Emp\_Designation, e.Emp\_Basic, e.Emp\_Gender, e.Emp\_Marital\_Status, "

+ "e.Emp\_Home\_Address, e.Emp\_Contact\_Num, e.Mgr\_Id "

+ "from Employee e inner join Department d "

+ "on e.Emp\_Dept\_ID=d.Dept\_ID "

+ "where e.Emp\_Dept\_ID = ? ";

public static final String FindByGrade="select e.Emp\_ID, e.Emp\_First\_Name, e.Emp\_Last\_Name, "

+ "e.Emp\_Date\_of\_Birth, e.Emp\_Date\_of\_Joining, e.Emp\_Dept\_ID, d.Dept\_Name, e.Emp\_Grade, "

+ "e.Emp\_Designation, e.Emp\_Basic, e.Emp\_Gender, e.Emp\_Marital\_Status, "

+ "e.Emp\_Home\_Address, e.Emp\_Contact\_Num, e.Mgr\_Id "

+ "from Employee e inner join Department d "

+ "on e.Emp\_Dept\_ID=d.Dept\_ID "

+ "where e.Emp\_Grade = ? ";

public static final String FindByMs="select e.Emp\_ID, e.Emp\_First\_Name, e.Emp\_Last\_Name, "

+ "e.Emp\_Date\_of\_Birth, e.Emp\_Date\_of\_Joining, e.Emp\_Dept\_ID,d.Dept\_Name, e.Emp\_Grade, "

+ "e.Emp\_Designation, e.Emp\_Basic, e.Emp\_Gender, e.Emp\_Marital\_Status, "

+ "e.Emp\_Home\_Address, e.Emp\_Contact\_Num, e.Mgr\_Id "

+ "from Employee e inner join Department d "

+ "on e.Emp\_Dept\_ID=d.Dept\_ID "

+ "where e.Emp\_Marital\_Status = ? ";

}













//DaoImpl



package com.cg.emp.dao;

import java.sql.Connection;

import java.sql.Date;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.text.Format;

import java.text.SimpleDateFormat;

import java.util.ArrayList;

import java.util.List;

import org.apache.log4j.Logger;

import org.apache.log4j.PropertyConfigurator;

import com.cg.emp.bean.Employee;

import com.cg.emp.exception.EmployeeException;

import com.cg.emp.util.DatabaseConnection;

public class EmployeeDaoImpl implements IEmployeeDao {

Logger logger=Logger.getRootLogger();

public EmployeeDaoImpl() {

PropertyConfigurator.configure("resources//log4j.properties");

}

Employee emp=null;

@Override

public String addEmp() throws EmployeeException {

return null;

}

@Override

public String modifyEmp() throws EmployeeException{

return null;

}

@Override

public List<Employee> dispEmp() throws EmployeeException

{

Connection con=DatabaseConnection.getConnection();

Format formatter = new SimpleDateFormat("yyyy-MM-dd");

List<Employee> list3=new ArrayList<Employee>();

Statement stmt=null;

try

{

stmt = con.createStatement();

ResultSet rs=stmt.executeQuery(QueryMapper.all);

while(rs.next())

{

Employee emp=new Employee();

emp.setEmpId(rs.getString("Emp\_ID"));

emp.setEmpFName(rs.getString("Emp\_First\_Name"));

emp.setEmpLName(rs.getString("Emp\_Last\_Name"));

Date d=rs.getDate("Emp\_Date\_of\_Birth");

emp.setEmpDOB(formatter.format(d));

Date d1=rs.getDate("Emp\_Date\_of\_Joining");

emp.setEmpDOJ(formatter.format(d1));

emp.setEmpDeptId(rs.getInt("Emp\_Dept\_ID"));

emp.setDept\_name(rs.getString("Dept\_Name"));

emp.setEmpGrade(rs.getString("Emp\_Grade"));

emp.setEmpDesign(rs.getString("Emp\_Designation"));

emp.setEmpBasic(rs.getInt("Emp\_Basic"));

emp.setEmpGender(rs.getString("Emp\_Gender"));

emp.setEmpMStatus(rs.getString("Emp\_Marital\_Status"));

emp.setEmpAddress(rs.getString("Emp\_Home\_Address"));

emp.setEmpContact(rs.getString("Emp\_Contact\_Num"));

emp.setEmpMgrId(rs.getString("Mgr\_Id"));

list3.add(emp);

}

rs.close();

con.close();

stmt.close();

}catch(SQLException e)

{

System.err.println(e.getMessage());

}

return list3;

}

@Override

public boolean applyLeave() throws EmployeeException{

return false;

}

@Override

public boolean approveLeave() throws EmployeeException {

return false;

}

@Override

public Employee searchEmpById(String empId) throws EmployeeException

{

Connection con=DatabaseConnection.getConnection();

Format formatter = new SimpleDateFormat("yyyy-MM-dd");

Employee emp=new Employee();

try

{

PreparedStatement ps=con.prepareStatement(QueryMapper.FindById);

ps.setString(1, empId);

ResultSet rs=ps.executeQuery();

if(rs.next())

{

emp.setEmpId(rs.getString("Emp\_ID"));

emp.setEmpFName(rs.getString("Emp\_First\_Name"));

emp.setEmpLName(rs.getString("Emp\_Last\_Name"));

Date d=rs.getDate("Emp\_Date\_of\_Birth");

emp.setEmpDOB(formatter.format(d));

Date d1=rs.getDate("Emp\_Date\_of\_Joining");

emp.setEmpDOJ(formatter.format(d1));

emp.setEmpDeptId(rs.getInt("Emp\_Dept\_ID"));

emp.setDept\_name(rs.getString("Dept\_Name"));

emp.setEmpGrade(rs.getString("Emp\_Grade"));

emp.setEmpDesign(rs.getString("Emp\_Designation"));

emp.setEmpBasic(rs.getInt("Emp\_Basic"));

emp.setEmpGender(rs.getString("Emp\_Gender"));

emp.setEmpMStatus(rs.getString("Emp\_Marital\_Status"));

emp.setEmpAddress(rs.getString("Emp\_Home\_Address"));

emp.setEmpContact(rs.getString("Emp\_Contact\_Num"));

emp.setEmpMgrId(rs.getString("Mgr\_Id"));

}

else

return null;

}catch(SQLException e)

{

System.err.println(e.getMessage());

}

return emp;

}

@Override

public List<Employee> searchEmpByFName(String empFName) throws EmployeeException

{

List<Employee> list=new ArrayList<Employee>();

Format formatter = new SimpleDateFormat("yyyy-MM-dd");

try

{

Connection con=DatabaseConnection.getConnection();

PreparedStatement ps=con.prepareStatement(QueryMapper.FindByFname);

ps.setString(1, empFName);

ResultSet rs=ps.executeQuery();

while(rs.next())

{

Employee emp = new Employee();

emp.setEmpId(rs.getString("Emp\_ID"));

emp.setEmpFName(rs.getString("Emp\_First\_Name"));

emp.setEmpLName(rs.getString("Emp\_Last\_Name"));

Date d=rs.getDate("Emp\_Date\_of\_Birth");

emp.setEmpDOB(formatter.format(d));

Date d1=rs.getDate("Emp\_Date\_of\_Joining");

emp.setEmpDOJ(formatter.format(d1));

emp.setEmpDeptId(rs.getInt("Emp\_Dept\_ID"));

emp.setDept\_name(rs.getString("Dept\_Name"));

emp.setEmpGrade(rs.getString("Emp\_Grade"));

emp.setEmpDesign(rs.getString("Emp\_Designation"));

emp.setEmpBasic(rs.getInt("Emp\_Basic"));

emp.setEmpGender(rs.getString("Emp\_Gender"));

emp.setEmpMStatus(rs.getString("Emp\_Marital\_Status"));

emp.setEmpAddress(rs.getString("Emp\_Home\_Address"));

emp.setEmpContact(rs.getString("Emp\_Contact\_Num"));

emp.setEmpMgrId(rs.getString("Mgr\_Id"));

list.add(emp);

}

rs.close();

con.close();

}catch(SQLException e)

{

System.err.println(e.getMessage());

}

return list;

}

@Override

public List<Employee> searchEmpByLName(String empLName) throws EmployeeException

{

List<Employee> list1=new ArrayList<Employee>();

Format formatter = new SimpleDateFormat("yyyy-MM-dd");

try

{

Connection con=DatabaseConnection.getConnection();

PreparedStatement ps=con.prepareStatement(QueryMapper.FindByLname);

ps.setString(1, empLName);

ResultSet rs=ps.executeQuery();

while(rs.next())

{

Employee emp = new Employee();

emp.setEmpId(rs.getString("Emp\_ID"));

emp.setEmpFName(rs.getString("Emp\_First\_Name"));

emp.setEmpLName(rs.getString("Emp\_Last\_Name"));

Date d=rs.getDate("Emp\_Date\_of\_Birth");

emp.setEmpDOB(formatter.format(d));

Date d1=rs.getDate("Emp\_Date\_of\_Joining");

emp.setEmpDOJ(formatter.format(d1));

emp.setEmpDeptId(rs.getInt("Emp\_Dept\_ID"));

emp.setDept\_name(rs.getString("Dept\_Name"));

emp.setEmpGrade(rs.getString("Emp\_Grade"));

emp.setEmpDesign(rs.getString("Emp\_Designation"));

emp.setEmpBasic(rs.getInt("Emp\_Basic"));

emp.setEmpGender(rs.getString("Emp\_Gender"));

emp.setEmpMStatus(rs.getString("Emp\_Marital\_Status"));

emp.setEmpAddress(rs.getString("Emp\_Home\_Address"));

emp.setEmpContact(rs.getString("Emp\_Contact\_Num"));

emp.setEmpMgrId(rs.getString("Mgr\_Id"));

list1.add(emp);

}

rs.close();

con.close();

}catch(SQLException e)

{

System.err.println(e.getMessage());

}

return list1;

}

@Override

public Employee searchEmpByDept(String dept) throws EmployeeException {

return null;

}

@Override

public List<Employee> searchEmpByGrade(String grade) throws EmployeeException

{

List<Employee> list2=new ArrayList<Employee>();

Format formatter = new SimpleDateFormat("yyyy-MM-dd");

try

{

Connection con=DatabaseConnection.getConnection();

PreparedStatement ps=con.prepareStatement(QueryMapper.FindByGrade);

ps.setString(1, grade);

ResultSet rs=ps.executeQuery();

while(rs.next())

{

Employee emp = new Employee();

emp.setEmpId(rs.getString("Emp\_ID"));

emp.setEmpFName(rs.getString("Emp\_First\_Name"));

emp.setEmpLName(rs.getString("Emp\_Last\_Name"));

Date d=rs.getDate("Emp\_Date\_of\_Birth");

emp.setEmpDOB(formatter.format(d));

Date d1=rs.getDate("Emp\_Date\_of\_Joining");

emp.setEmpDOJ(formatter.format(d1));

emp.setEmpDeptId(rs.getInt("Emp\_Dept\_ID"));

emp.setDept\_name(rs.getString("Dept\_Name"));

emp.setEmpGrade(rs.getString("Emp\_Grade"));

emp.setEmpDesign(rs.getString("Emp\_Designation"));

emp.setEmpBasic(rs.getInt("Emp\_Basic"));

emp.setEmpGender(rs.getString("Emp\_Gender"));

emp.setEmpMStatus(rs.getString("Emp\_Marital\_Status"));

emp.setEmpAddress(rs.getString("Emp\_Home\_Address"));

emp.setEmpContact(rs.getString("Emp\_Contact\_Num"));

emp.setEmpMgrId(rs.getString("Mgr\_Id"));

list2.add(emp);

}

rs.close();

con.close();

}catch(SQLException e)

{

System.err.println(e.getMessage());

}

return list2;

}

@Override

public Employee searchEmpByMS(String ms) throws EmployeeException{

return null;

}

}







//Client



package com.cg.emp.ui;

import java.util.ArrayList;

import java.util.Iterator;

import java.util.List;

import java.util.Scanner;

import com.cg.emp.bean.Employee;

import com.cg.emp.exception.EmployeeException;

import com.cg.emp.service.EmployeeServImpl;

import com.cg.emp.service.IEmployeeServ;

public class Home {

public static void main(String[] args)

{

Scanner sc=new Scanner(System.in);

IEmployeeServ service=new EmployeeServImpl();

Employee e1=new Employee();

int choice=0,empCh=0,adminCh=0;

String eid, fname, lname, grade;

while(true) {

System.out.println("Enter ur choice:\n1. Admin\n2. Employee\n3. Exit");

choice=sc.nextInt();

switch(choice)

{

case 1://for Admin

System.out.println("\*\*\*\*\*\*\*For Admin\*\*\*\*\*\*\*\*");

while(true)

{

System.out.println("\nEnter Choice\n1. Add Employee\n2. Modify Employee\n3. Approve Leave\n4. Display all Employee\n5. Exit");

adminCh=sc.nextInt();

switch(adminCh)

{

case 1://add employee

break;

case 2://modify employee

break;

case 3://approve leave

break;

case 4://display all employee

List<Employee> l3= new ArrayList<Employee>();

try {

l3=service.dispEmp();

if (l3.isEmpty())

System.out.println("Sorry...Record Not Found...!!!");

else

{

Iterator<Employee> i = l3.iterator();

System.out.println("Employee Details:-");

System.out.println("-----------------------");

System.out.format("%s%15s%15s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s\n","Emp ID","First Name","Last Name","DOB","DOJ","Dept ID","Dept Name","Grade","Designation","Basic","Gender","Marital Status","Address","Contact","Manager ID");

while (i.hasNext())

{

Employee e = (Employee) i.next();

System.out.format("%s%10s%10s%20s%20s%10d%10s%10s%10s%10d%10s%10s%10s%10s%10s\n",e.getEmpId(),e.getEmpFName(),e.getEmpLName(),e.getEmpDOB(),e.getEmpDOJ(),e.getEmpDeptId(),e.getDept\_name(),e.getEmpGrade(),e.getEmpDesign(),e.getEmpBasic(),e.getEmpGender(),e.getEmpMStatus(),e.getEmpAddress(),e.getEmpContact(),e.getEmpMgrId());

}

}

} catch (EmployeeException e) {

System.err.println(e.getMessage());

}

break;

case 5://exit

System.exit(0);

default:

System.out.println("Invalid Choice....");

}

}

case 2://for Employee

System.out.println("\*\*\*\*\*\*\*For Employee\*\*\*\*\*\*\*");

while(true)

{

System.out.println("\nEnter choice\n1. Search by Employee id\n2. Search by First Name\n3. Search by Last Name"

+ "\n4. Search by Department\n5. Search by Grade\n6. Search by Marital Status\n7. Apply for leave\n8. Exit");

empCh=sc.nextInt();

switch(empCh)

{

case 1://Search by Employee Id

System.out.println("Enter Employee ID: ");

eid=sc.next();

try {

e1=service.searchEmpById(eid);

if(e1==null)

{

System.out.println("Sorry...Record Not Found...!!!");

}

else

{

System.out.println("Employee ID : "+e1.getEmpId()+"\nFirst Name : "+e1.getEmpFName()+"\nLast Name : "+e1.getEmpLName()+"\nDate of Birth : "+e1.getEmpDOB()+"\nDate of Joining : "+e1.getEmpDOJ()+""

+ "\nDepartment ID : "+e1.getEmpDeptId()+"\nDepartment Name : "+e1.getDept\_name()+"\nGrade : "+e1.getEmpGrade()+"\nDesignation : "+e1.getEmpDesign()+"\nEmployee Basic : "+e1.getEmpBasic()+"\nGender : "+e1.getEmpGender()+"\nMarital Status : "+e1.getEmpMStatus()+"\nHome Address : "+e1.getEmpAddress()+"\nContact Number : "+e1.getEmpContact()+"\nEmployee's Manager ID : "+e1.getEmpMgrId());

}

} catch (EmployeeException e) {

System.err.println(e.getMessage());

}

break;

case 2://Search by first name

System.out.println("Enter First Name: ");

fname=sc.next();

List<Employee> l1= new ArrayList<Employee>();

try {

l1=service.searchEmpByFName(fname);

if (l1.isEmpty())

System.out.println("Sorry...Record Not Found...!!!");

else

{

Iterator<Employee> i = l1.iterator();

System.out.println("Employee Details:-");

System.out.println("-----------------------");

System.out.format("%s%15s%15s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s\n","Emp ID","First Name","Last Name","DOB","DOJ","Dept ID","Dept Name","Grade","Designation","Basic","Gender","Marital Status","Address","Contact","Manager ID");

while (i.hasNext())

{

Employee e = (Employee) i.next();

System.out.format("%s%10s%10s%20s%20s%10d%10s%10s%10s%10d%10s%10s%10s%10s%10s\n",e.getEmpId(),e.getEmpFName(),e.getEmpLName(),e.getEmpDOB(),e.getEmpDOJ(),e.getEmpDeptId(),e.getDept\_name(),e.getEmpGrade(),e.getEmpDesign(),e.getEmpBasic(),e.getEmpGender(),e.getEmpMStatus(),e.getEmpAddress(),e.getEmpContact(),e.getEmpMgrId());

}

}

} catch (EmployeeException e) {

System.err.println(e.getMessage());

}

break;

case 3://Search by last name

System.out.println("Enter Last Name: ");

lname=sc.next();

List<Employee> l12= new ArrayList<Employee>();

try {

l12=service.searchEmpByLName(lname);

if (l12.isEmpty())

System.out.println("Sorry...Record Not Found...!!!");

else

{

Iterator<Employee> i = l12.iterator();

System.out.println("Employee Details:-");

System.out.println("-----------------------");

System.out.format("%s%15s%15s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s\n","Emp ID","First Name","Last Name","DOB","DOJ","Dept ID","Dept Name","Grade","Designation","Basic","Gender","Marital Status","Address","Contact","Manager ID");

while (i.hasNext())

{

Employee e = (Employee) i.next();

System.out.format("%s%10s%10s%20s%20s%10d%10s%10s%10s%10d%10s%10s%10s%10s%10s\n",e.getEmpId(),e.getEmpFName(),e.getEmpLName(),e.getEmpDOB(),e.getEmpDOJ(),e.getEmpDeptId(),e.getDept\_name(),e.getEmpGrade(),e.getEmpDesign(),e.getEmpBasic(),e.getEmpGender(),e.getEmpMStatus(),e.getEmpAddress(),e.getEmpContact(),e.getEmpMgrId());

}

}

} catch (EmployeeException e) {

System.err.println(e.getMessage());

}

break;

case 4://Search by Department

break;

case 5://Search by grade

System.out.println("Enter Grade: ");

grade=sc.next();

List<Employee> l13= new ArrayList<Employee>();

try {

l13=service.searchEmpByGrade(grade);

if (l13.isEmpty())

System.out.println("Sorry...Record Not Found...!!!");

else

{

Iterator<Employee> i = l13.iterator();

System.out.println("Employee Details:-");

System.out.println("-----------------------");

System.out.format("%s%15s%15s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s%10s\n","Emp ID","First Name","Last Name","DOB","DOJ","Dept ID","Dept Name","Grade","Designation","Basic","Gender","Marital Status","Address","Contact","Manager ID");

while (i.hasNext())

{

Employee e = (Employee) i.next();

System.out.format("%s%10s%10s%20s%20s%10d%10s%10s%10s%10d%10s%10s%10s%10s%10s\n",e.getEmpId(),e.getEmpFName(),e.getEmpLName(),e.getEmpDOB(),e.getEmpDOJ(),e.getEmpDeptId(),e.getDept\_name(),e.getEmpGrade(),e.getEmpDesign(),e.getEmpBasic(),e.getEmpGender(),e.getEmpMStatus(),e.getEmpAddress(),e.getEmpContact(),e.getEmpMgrId());

}

}

} catch (EmployeeException e) {

System.err.println(e.getMessage());

}

break;

case 6://Search by Martial Status

break;

case 7://Apply for leave

break;

case 8://exit

System.exit(0);

default:

System.out.println("Invalid Choice....");

break;

}

}

case 3://exit

System.exit(0);

default:

System.out.println("Invalid Choice....");

break;

}

}

}

}























































//dao



@Override

public Employee searchEmpById(String empId) throws EmployeeException

{

Connection con=DatabaseConnection.getConnection();

Format formatter = new SimpleDateFormat("yyyy-MM-dd");

Employee emp=new Employee();

try

{

PreparedStatement ps=con.prepareStatement(QueryMapper.FindById);

ps.setString(1, empId);

ResultSet rs=ps.executeQuery();

if(rs.next())

{

emp.setEmpId(rs.getString("Emp\_ID"));

emp.setEmpFName(rs.getString("Emp\_First\_Name"));

emp.setEmpLName(rs.getString("Emp\_Last\_Name"));

Date d=rs.getDate("Emp\_Date\_of\_Birth");

emp.setEmpDOB(formatter.format(d));

Date d1=rs.getDate("Emp\_Date\_of\_Joining");

emp.setEmpDOJ(formatter.format(d1));

emp.setEmpDeptId(rs.getInt("Emp\_Dept\_ID"));

emp.setDept\_name(rs.getString("Dept\_Name"));

emp.setEmpGrade(rs.getString("Emp\_Grade"));

emp.setEmpDesign(rs.getString("Emp\_Designation"));

emp.setEmpBasic(rs.getInt("Emp\_Basic"));

emp.setEmpGender(rs.getString("Emp\_Gender"));

emp.setEmpMStatus(rs.getString("Emp\_Marital\_Status"));

emp.setEmpAddress(rs.getString("Emp\_Home\_Address"));

emp.setEmpContact(rs.getString("Emp\_Contact\_Num"));

emp.setEmpMgrId(rs.getString("Mgr\_Id"));

}

else

return null;

}catch(SQLException e)

{

System.err.println(e.getMessage());

}

return emp;

}





//client









package com.cg.emp.ui;

import java.util.ArrayList;

import java.util.List;

import java.util.Scanner;

import com.cg.emp.bean.Employee;

import com.cg.emp.exception.EmployeeException;

import com.cg.emp.service.EmployeeServImpl;

import com.cg.emp.service.IEmployeeServ;

public class Home {

public static void main(String[] args)

{

Scanner sc=new Scanner(System.in);

IEmployeeServ service=new EmployeeServImpl();

int choice=0,empCh=0;

String eid, fname, lname, dept, grade, ms;

while(true) {

System.out.println("Enter ur choice:\n1. Admin\n2. Employee\n3. Exit");

choice=sc.nextInt();

switch(choice)

{

case 1:System.out.println("\*\*\*\*\*\*\*For Admin\*\*\*\*\*\*\*\*");

break;

case 2:System.out.println("\*\*\*\*\*\*\*For Employee\*\*\*\*\*\*\*");

while(true)

{

System.out.println("\nEnter choice for search\n1. Search by Employee id\n2. Search by First Name\n3. Search by Last Name"

+ "\n4. Search by Department\n5. Search by Grade\n6. Search by Marital Status\n7. Apply for leave");

empCh=sc.nextInt();

switch(empCh)

{

case 1:

System.out.println("Enter Employee ID: ");

eid=sc.next();

Employee e=new Employee();

try {

e=service.searchEmpById(eid);

if(e==null)

{

System.out.println("Sorry...Record Not Found...!!!");

}

else

{

System.out.println("Employee ID : "+e.getEmpId()+"\nFirst Name : "+e.getEmpFName()+"\nLast Name : "+e.getEmpLName()+"\nDate of Birth : "+e.getEmpDOB()+"\nDate of Joining : "+e.getEmpDOJ()+""

+ "\nDepartment ID : "+e.getEmpDeptId()+"\nDepartment Name : "+e.getDept\_name()+"\nGrade : "+e.getEmpGrade()+"\nDesignation : "+e.getEmpDesign()+"\nEmployee Basic : "+e.getEmpBasic()+"\nGender : "+e.getEmpGender()+"\nMarital Status : "+e.getEmpMStatus()+"\nHome Address : "+e.getEmpAddress()+"\nContact Number : "+e.getEmpContact()+"\nEmployee's Manager ID : "+e.getEmpMgrId());

}

} catch (EmployeeException e1) {

System.err.println(e1.getMessage());

}

break;

case 2:

System.out.println("Enter First Name: ");

fname=sc.next();

List<Employee> l1= new ArrayList<Employee>();

try {

l1=service.searchEmpByFName(fname);

if(l1==null)

{

System.out.println("Sorry...Record Not Found...!!!");

}

else

{

for(Employee e1 : l1) {

System.out.println(e1);

}

}

} catch (EmployeeException e1) {

System.err.println(e1.getMessage());

}

break;

case 3:

System.out.println("Enter Last Name: ");

lname=sc.next();

try {

e=service.searchEmpByLName(lname);

if(e==null)

{

System.out.println("Sorry...Record Not Found...!!!");

}

else

{

System.out.println("Employee ID : "+e.getEmpId()+"\nFirst Name : "+e.getEmpFName()+"\nLast Name : "+e.getEmpLName()+"\nDate of Birth : "+e.getEmpDOB()+"\nDate of Joining : "+e.getEmpDOJ()+""

+ "\nDepartment ID : "+e.getEmpDeptId()+"\nGrade : "+e.getEmpGrade()+"\nDesignation : "+e.getEmpDesign()+"\nEmployee Basic : "+e.getEmpBasic()+"\nGender : "+e.getEmpGender()+"\nMarital Status : "+e.getEmpMStatus()+"\nHome Address : "+e.getEmpAddress()+"\nContact Number : "+e.getEmpContact()+"\nEmployee's Manager ID : "+e.getEmpMgrId());

}

} catch (EmployeeException e1) {

System.err.println(e1.getMessage());

}

break;

case 4:

break;

case 5:

System.out.println("Enter Grade : ");

grade=sc.next();

try {

e=service.searchEmpByGrade(grade);

if(e==null)

{

System.out.println("Sorry...Record Not Found...!!!");

}

else

{

System.out.println("Employee ID : "+e.getEmpId()+"\nFirst Name : "+e.getEmpFName()+"\nLast Name : "+e.getEmpLName()+"\nDate of Birth : "+e.getEmpDOB()+"\nDate of Joining : "+e.getEmpDOJ()+""

+ "\nDepartment ID : "+e.getEmpDeptId()+"\nGrade : "+e.getEmpGrade()+"\nDesignation : "+e.getEmpDesign()+"\nEmployee Basic : "+e.getEmpBasic()+"\nGender : "+e.getEmpGender()+"\nMarital Status : "+e.getEmpMStatus()+"\nHome Address : "+e.getEmpAddress()+"\nContact Number : "+e.getEmpContact()+"\nEmployee's Manager ID : "+e.getEmpMgrId());

}

} catch (EmployeeException e1) {

System.err.println(e1.getMessage());

}

break;

case 6:

break;

default:

break;

}

}

case 3:System.exit(0);

break;

default:

break;

}

}

}

}





//Nevil trainee app



https://fts.capgemini.com/pubpwd/41748228908352/TraineeApp.zip

link valid until 2018-08-06 11:00 UTC (public access with local password)

Additional credentials, to enter in specific popup box when prompted for authentication.

username: ukhydaaiq

Password: P8KhGTDZg8















//Trainee app

https://fts.capgemini.com/pubpwd/9185377902690/Spring\_Lab2.zip link valid until 2018-08-06 11:00 UTC (public access with local password)

Additional credentials, to enter in specific popup box when prompted for authentication.

username: oteoonkzl

Password: 4tK2PyVPEM













\\\\\\\\\\product details\\\\\\\\\\





The file(s) can be downloaded with the following URL(s) and credentials:

https://fts.capgemini.com/pubpwd/15725150502117/ProductDetail.7z link valid until 2018-08-01 6:00 UTC (public access with local password)

Additional credentials, to enter in specific popup box when prompted for authentication.

username: vwyeuodmg

Password: cqDN2W9sdZ







]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]]













-----------------WILDFLY------------------------





The file(s) can be downloaded with the following URL(s) and credentials:

https://fts.capgemini.com/pubpwd/42119093965655/wildfly-9.0.1.Final.zip link valid until 2018-07-31 13:00 UTC (public access with local password)

Additional credentials, to enter in specific popup box when prompted for authentication.

username: clnmzyqqg

Password: qNk5t67FRU







---------------------------------------------------------------------------------------------------------------------------













The file(s) can be downloaded with the following URL(s) and credentials:

https://fts.capgemini.com/pubpwd/20909170972680/ProductApplication.zip link valid until 2018-07-31 10:00 UTC (public access with local password)

Additional credentials, to enter in specific popup box when prompted for authentication.

username: eeyolgxgm

Password: GzxEa8nhQP





-------------------------------------------------------------------------------------------------------





Password: KYzwmkTCYD

username: pireraucm

https://fts.capgemini.com/pubpwd/37329849332328/Trainee.7z





-------------------------------------------------------------------------------------------------------









































"Which operator is used to work with class scoped methods/scopes

" "p()

" T() v() D() B

When serializing an object to a file, the standard convention in Java is to give the file a: .war .java .ser none of the mentioned C

Which is the central component of Spring MVC? Controller dispatcher Servelet both a nd b none of the mentioned B

If a bean can be created any number of times, scope is session prototype global-session request B

Which of the following annotation of JAX RS API binds the parameter passed to method to a value in path? MatrixParam ParamPath QueryParam annotation  PathParam annotation D

"Script’s location is specified by attribute

" script-loc script-source script-annotation none of the mentioned B

Spring HATEOS is? "

Link creation and representation assembly " Software architecture dsign pattern New framework to design bootstrap Reduces size of build file A

"Spring-WS provides various abstract endpoint classes for you to process the request.

" "org.springframework.ws.server.endpoint

 " org.springframework.ws.server org.springframework.\* none of the mentioned A

Attribute used to handle web flow requests. servlet-mapping servlet-attr servlet-flow servlet-requests A







--------------------GIT----------------------------

















meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1 (master)

$ git init B

Initialized empty Git repository in D:/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B/.git/

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1 (master)

$ git init C

Initialized empty Git repository in D:/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C/.git/

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1 (master)

$ git init D

Initialized empty Git repository in D:/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D/.git/

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1 (master)

$ cd B

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git add info.txt

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git commit -m "committed"

[master (root-commit) 3a6779e] committed

Committer: Patil <meghana.patil@capgemini.com>

Your name and email address were configured automatically based

on your username and hostname. Please check that they are accurate.

You can suppress this message by setting them explicitly. Run the

following command and follow the instructions in your editor to edit

your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 1 insertion(+)

create mode 100644 info.txt

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git checkout Development

error: pathspec 'Development' did not match any file(s) known to git.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git checkout master

Already on 'master'

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git checkout -b Development

Switched to a new branch 'Development'

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Development)

$ git checkout master

Switched to branch 'master'

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git checkout -b Testing

Switched to a new branch 'Testing'

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Testing)

$ git remote add origin https://github.com/meghanapatil25/A.git

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Testing)

$ git push -u origin A

error: src refspec A does not match any.

error: failed to push some refs to 'https://github.com/meghanapatil25/A.git'

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Testing)

$ git push -u originA A

error: src refspec A does not match any.

error: failed to push some refs to 'originA'

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Testing)

$ git push origin A

error: src refspec A does not match any.

error: failed to push some refs to 'https://github.com/meghanapatil25/A.git'

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Testing)

$ git push -u origin master

Logon failed, use ctrl+c to cancel basic credential prompt.

Username for 'https://github.com/':

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Testing)

$ git push -u origin --all

Logon failed, use ctrl+c to cancel basic credential prompt.

Username for 'https://github.com/':

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Testing)

$

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Testing)

$ git remote add originA https://github.com/meghanapatil25/A.git

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (Testing)

$ git checkout master

Switched to branch 'master'

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git remote add originA https://github.com/meghanapatil25/A.git

fatal: remote originA already exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git push -u origin --all

Logon failed, use ctrl+c to cancel basic credential prompt.

Username for 'https://github.com/':

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git push -u originA --all

Logon failed, use ctrl+c to cancel basic credential prompt.

Username for 'https://github.com/':

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git push -u originA Development

Counting objects: 3, done.

Writing objects: 100% (3/3), 215 bytes | 0 bytes/s, done.

Total 3 (delta 0), reused 0 (delta 0)

error: waitpid for send-pack failed: No child processes

To https://github.com/meghanapatil25/A.git

\* [new branch] Development -> Development

Branch Development set up to track remote branch Development from originA.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git push -u originA Testing

Total 0 (delta 0), reused 0 (delta 0)

error: waitpid for send-pack failed: No child processes

To https://github.com/meghanapatil25/A.git

\* [new branch] Testing -> Testing

Branch Testing set up to track remote branch Testing from originA.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ git checkout master

Already on 'master'

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$ ^C

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/B (master)

$

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ git pull originA C

fatal: 'originA' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ git pull originA Development

fatal: 'originA' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ git remote add originA https://github.com/meghanapatil25/A.git

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ git pull originA Development

remote: Counting objects: 3, done.

remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0

Unpacking objects: 100% (3/3), done.

error: waitpid for fetch-pack failed: No child processes

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ git pull originA Development

From https://github.com/meghanapatil25/A

\* branch Development -> FETCH\_HEAD

\* [new branch] Development -> originA/Development

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ git add MyJavaCode.txt

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ git commit -m "Committed C"

[master d408f62] Committed C

Committer: Patil <meghana.patil@capgemini.com>

Your name and email address were configured automatically based

on your username and hostname. Please check that they are accurate.

You can suppress this message by setting them explicitly. Run the

following command and follow the instructions in your editor to edit

your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 MyJavaCode.txt

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ ^C

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ git pull originA

remote: Counting objects: 7, done.

remote: Compressing objects: 100% (4/4), done.

remote: Total 7 (delta 2), reused 7 (delta 2), pack-reused 0

Unpacking objects: 100% (7/7), done.

error: waitpid for fetch-pack failed: No child processes

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$ git pull originA

From https://github.com/meghanapatil25/A

\* [new branch] Testing -> originA/Testing

\* [new branch] master -> originA/master

You asked to pull from the remote 'originA', but did not specify

a branch. Because this is not the default configured remote

for your current branch, you must specify a branch on the command line.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C (master)

$

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git remote add origin https://github.com/meghanapatil25/MyGitDemo.git

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull origin Testing

remote: Counting objects: 3, done.

remote: Total 3 (delta 0), reused 3 (delta 0), pack-reused 0

Unpacking objects: 100% (3/3), done.

error: waitpid for fetch-pack failed: No child processes

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull origin Testing

From https://github.com/meghanapatil25/MyGitDemo

\* branch Testing -> FETCH\_HEAD

\* [new branch] Testing -> origin/Testing

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git add MyJUnitTestCase.txt

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git commit -m "Committed D"

[master 3503582] Committed D

Committer: Patil <meghana.patil@capgemini.com>

Your name and email address were configured automatically based

on your username and hostname. Please check that they are accurate.

You can suppress this message by setting them explicitly. Run the

following command and follow the instructions in your editor to edit

your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 MyJUnitTestCase.txt

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull C

fatal: 'C' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull D:\MeghanaPatil\_153231\DevOps\_assignments\Lab1\C

fatal: 'D:MeghanaPatil\_153231DevOps\_assignmentsLab1C' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull D:\MeghanaPatil\_153231\DevOps\_assignments\Lab1\C

fatal: 'D:MeghanaPatil\_153231DevOps\_assignmentsLab1C' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull \DevOps\_assignments\Lab1\C

fatal: 'DevOps\_assignmentsLab1C' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull D:\MeghanaPatil\_153231\DevOps\_assignments\Lab1\C

fatal: 'D:MeghanaPatil\_153231DevOps\_assignmentsLab1C' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ ^C

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ ^C

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ ^C

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git config --edit

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git config --edit MyJavaCode.txt

error: wrong number of arguments

usage: git config [<options>]

Config file location

--global use global config file

--system use system config file

--local use repository config file

-f, --file <file> use given config file

--blob <blob-id> read config from given blob object

Action

--get get value: name [value-regex]

--get-all get all values: key [value-regex]

--get-regexp get values for regexp: name-regex [value-regex]

--get-urlmatch get value specific for the URL: section[.var] URL

--replace-all replace all matching variables: name value [value\_regex]

--add add a new variable: name value

--unset remove a variable: name [value-regex]

--unset-all remove all matches: name [value-regex]

--rename-section rename section: old-name new-name

--remove-section remove a section: name

-l, --list list all

-e, --edit open an editor

--get-color find the color configured: slot [default]

--get-colorbool find the color setting: slot [stdout-is-tty]

Type

--bool value is "true" or "false"

--int value is decimal number

--bool-or-int value is --bool or --int

--path value is a path (file or directory name)

Other

-z, --null terminate values with NUL byte

--name-only show variable names only

--includes respect include directives on lookup

--show-origin show origin of config (file, standard input, blob, command line)

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git add MyJavaCode.txt

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git commit -m "Committed MyJavaCode"

[master d95f9ac] Committed MyJavaCode

Committer: Patil <meghana.patil@capgemini.com>

Your name and email address were configured automatically based

on your username and hostname. Please check that they are accurate.

You can suppress this message by setting them explicitly. Run the

following command and follow the instructions in your editor to edit

your configuration file:

git config --global --edit

After doing this, you may fix the identity used for this commit with:

git commit --amend --reset-author

1 file changed, 1 insertion(+)

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git push -u originA --all

fatal: 'originA' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git push -u origin --all

Counting objects: 10, done.

Delta compression using up to 4 threads.

Compressing objects: 100% (8/8), done.

Writing objects: 100% (10/10), 941 bytes | 0 bytes/s, done.

Total 10 (delta 3), reused 0 (delta 0)

remote: Resolving deltas: 100% (3/3), done.

remote: This repository moved. Please use the new location:

remote: https://github.com/meghanapatil25/A.git

error: waitpid for send-pack failed: No child processes

To https://github.com/meghanapatil25/MyGitDemo.git

\* [new branch] master -> master

Branch master set up to track remote branch master from origin.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull D:\MeghanaPatil\_153231\DevOps\_assignments\Lab1\C

fatal: 'D:MeghanaPatil\_153231DevOps\_assignmentsLab1C' does not appear to be a git repository

fatal: Could not read from remote repository.

Please make sure you have the correct access rights

and the repository exists.

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull --all

Fetching origin

From https://github.com/meghanapatil25/MyGitDemo

\* [new branch] Development -> origin/Development

There is no tracking information for the current branch.

Please specify which branch you want to merge with.

See git-pull(1) for details.

git pull <remote> <branch>

If you wish to set tracking information for this branch you can do so with:

git branch --set-upstream-to=origin/<branch> master

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C

remote: Counting objects: 3, done.

remote: Compressing objects: 100% (2/2), done.

remote: Total 3 (delta 0), reused 0 (delta 0)

error: waitpid for pack-objects failed: No child processes

error: git upload-pack: git-pack-objects died with error.

fatal: git uploremote: aborting due to possible repository corruption on the remote side.

ad-pack: aborting due to possible repository corruption on the remote side.

Unpacking objects: 100% (3/3), done.

fatal: error in sideband demultiplexer

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git pull /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C

From D:/MeghanaPatil\_153231/DevOps\_assignments/Lab1/C

\* branch HEAD -> FETCH\_HEAD

Merge made by the 'recursive' strategy.

MyJavaCode.txt | 0

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 MyJavaCode.txt

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$ git status

On branch master

Your branch is up-to-date with 'origin/master'.

nothing to commit, working tree clean

meghanpa@DIN51002648 MINGW64 /d/MeghanaPatil\_153231/DevOps\_assignments/Lab1/D (master)

$







