Week3 Revisit

**Day 11** **Generics, Annotations, Comparable & Comparator** [Compare Objects]

a=15, b=25; a>b = false, a<b= true; a==b false; a!=b true; (Comparing objects )

Annotations are meta data, which gives more info about the code to the JVM.

Annotations always starts with @ symbol. Can be applied to class, interface, methods and variables, & parameters.

Annotations replaces xml configurations

**Day 12** **Threads, Garbage collection, StringBuffer & StringBuilder, Heap & Stack**

Threads enables parallel programming. It’s used to perform multiple operations at the same time.

Creating Threads (Thread class, Runnable Interface) , start(), run(), join(), stop(), yield(), sleep() etc.,

States of Thread , NEW, RUNNABLE, RUNNING, NON-RUNNING, TERMINATED

System.gc() 🡪 Calls the Garbage collector. While doing gc, jvm will call the finalize() method.

Garbage collection is the process of reclaiming memory space used by un-referenced object.

**Day 13** **Functional Interfaces, Lambdas, Maven**

Functional Interface – Is a type of Interface where we have only one abstract method.

Functional Interface can have many default and static methods but only one abstract method.

Functional interface implementation can be given by Lambdas.

Lambda means anonymous methods. ()🡪 {body;};

Lambda is also called as One line function.

Calling the lambda, can be done with the help of Functional Interface.

In java8, there is a new annotation added “@FunctionalInterface”

**Day 14 Maven Life Cycle, Repo (Local repo & Remote repo)**

Maven Commands (mvn archetype:generate, mvn install, mvn clean, mvn test, mvn package)

Maven is Project Management tool. It build, generates, test, package, deploy and manage the project dependencies….

**Day 15 IO Operations using Streams, Serialization & Logging**

Streams & Reader/Writter

Streams are byte based ( 1 byte only) – InputStream, OutputStream, BufferedStream, ObjectStream, FileStream.

Reader/Writer are character based (2 bytes) – FileReader/FileWritter, File, Scanner

File System – In side your operating system &/ hard disk. (txt, .doc, pdf, jpeg, png )

Serialization – Is the process of storing the state of an object In a flat file system. (txt,rtf,doc)

The reverse of serialization process is called de-serialization.

(By reading the content of a flat file, generating objects using this data)

Files type

1. Flat file (Simple/Normal Text file)
2. Formatted file (Structured/ organized file - -database table, excel sheet

Logging – Is the process of storing the outputs of console.

Log levels – INFO, WARN, ERROR, DEBUG

Log4j, slf4j

Week4 – RDBMS (Relational Database Management System)

Data – Useful Information

Siva – (Information)

Person Name = Siva (Data)

Database – Is the way of storing & retrieving the data in an organized format.

1. Easy to store & Easy to read, search, filter

DBMS – Database Management System. (Storing the data in Table row & column format)

Two types of Relationship in JAVA

1. Is A & Has A (extends & implements) Car is a Vehicle, Dog is an Animal, Manager is a Employee, Car has a engine, Dog has a Tail, [ Specifying the relationship between two or more classes, interfaces and java programs

Types of relationship in RDBMS

1. One to One Relationship (primary key 1..1) – One person can have only one passport, one person can have only one SSN)
2. Many to One/One to Many (1..N) [One student can write multiple exams]
3. Many to Many (M..N) – child table/ derived table [Session can have multiple students. Students can join in to different sessions.

AWS – Amazon Web Service

Many Regional Centers & Availability Zone.

AWS (Personnel ID – Personal Free tier account.

Logging to AWS console window. (Even though it’s a free tier, depending upon the usage, the account will be charged)

Creating AWS Free Tier account.

<https://github.com/syskantechnosoft/revature_batch1/blob/main/AWS%20RDS%20Free%20Tier.txt>

<https://github.com/syskantechnosoft/revature_batch1/blob/main/AWS%20RDS%20Setup.docx>

MySQL Installer for Windows - <https://dev.mysql.com/downloads/installer/>

**CREATE** **TABLE** public.employee (

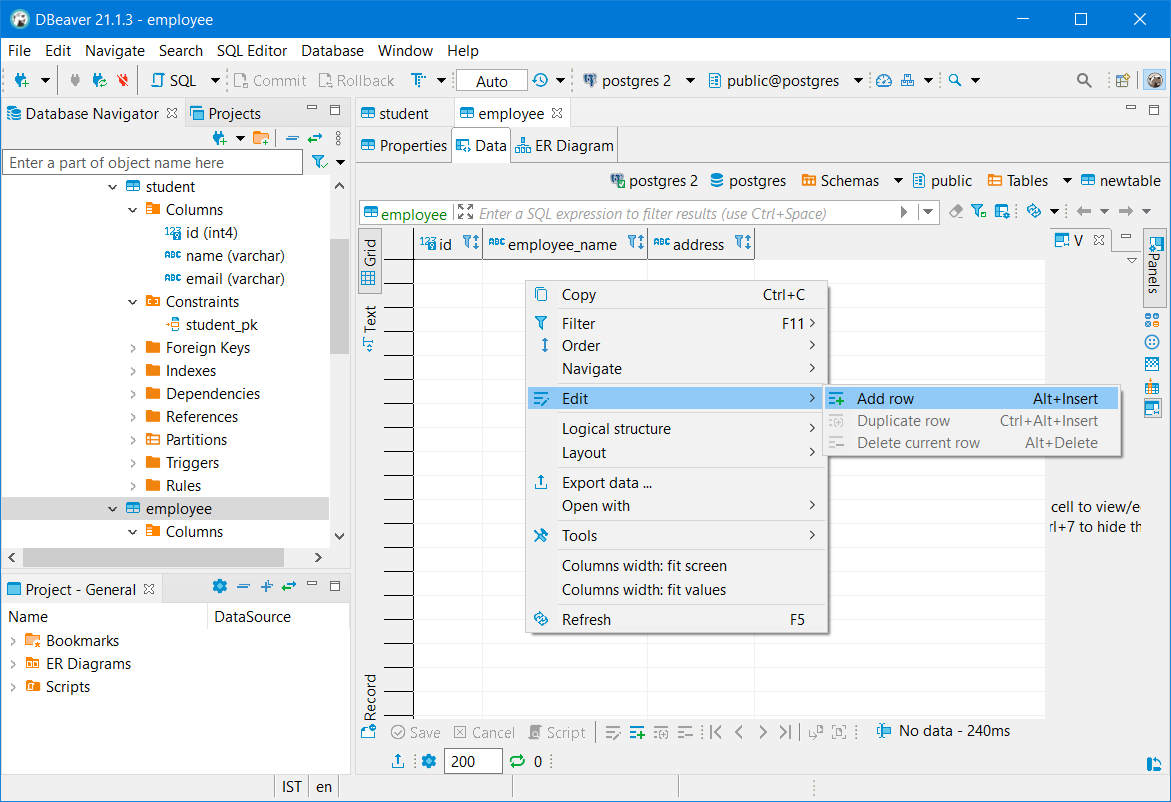
id **int4** **NOT** **NULL**,

employee\_name **varchar** **NULL**,

address **varchar** **NULL**,

**CONSTRAINT** employee\_pk **PRIMARY** **KEY** (id)

);



Amazon RDS – Relational Database Service (MySQL, Postgres, MS-SQL) Create a database in cloud.

Types of Database

1. SQL based
   1. Stand-alone (Oracle, MySQL, MS-SQL, Postgres, MariaDB, AuroraDB, DB2)
   2. In-memory database (h2, elastic cache, derby, sqlite)
2. No-SQL Database
   1. Document based (Mongo DB, Redis, Cassandra)
   2. Graph based (Neo-SQL, Neptune)

Amazon RDS CLI Commands - <https://docs.aws.amazon.com/cli/latest/reference/rds/index.html>

SQL – Structured Query Language

SQL – is the Language for the Relational Database

In Database, the data will be organized into tables. Collection of rows & columns.

Each row represents detail about a particular person, place or thing.

Each column represents a particular property for all (person/place/thing)

Relational – It’s bcos using Mathematics relation theory (Set, intersect, union…)