

Blastonbury Pop Festival – 5COM2005 (Database Systems)

Part 1 – ORACLE (Marks – 50)

Q1.1 Database Modelling / SQL (marks -10)

Entities:

There are 5 entities for the given case study: -

1. Agent
2. Team
3. Members
4. Team_Performance
5. Stage

Agent– This entity contains four attributes –Agent_Id , which display the unique agent id number, Agent_Name which shows the name of Agent , phone which shows the phone number of agent , then Email which displays the mail id of agent and Agent_Role which displays the role of agent.

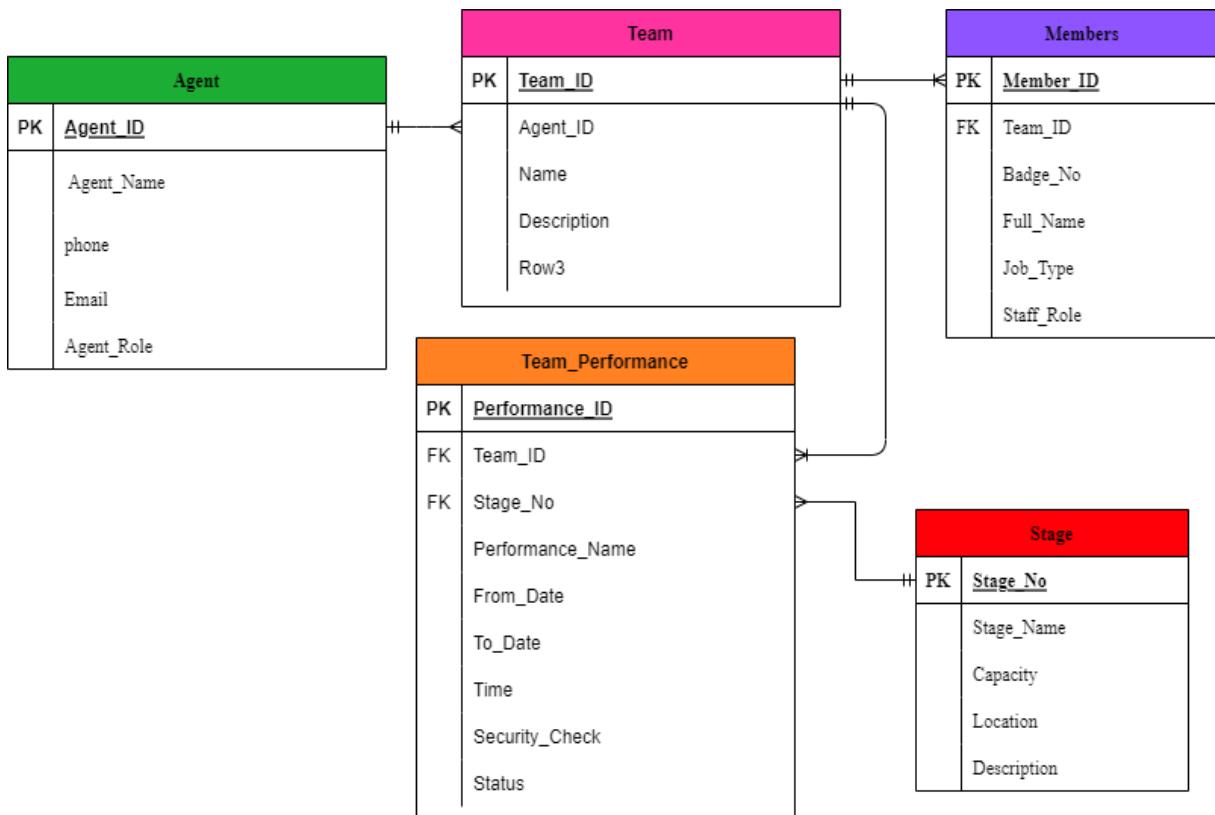
Team – This entity contains four attributes – Team_ID which displays the unique team id ,Agent_Id which shows the unique agent id, Name which displays for name of team , then description which express the full description about Team and row3.

Members –This entity contains 6 attributes named – Member_id which displays the unique member id , Team_Id which displays the unique team id , Badge_No which displays the badge number , Full_Name which displays the member full name , Job_Type which displays the type of job and Staff_role which displays the role of staff.

Team_Performance – This entity contains 9 attributes – unique Performance_id , Team_id which shows the unique team id , stage_no. which shows the stage number , thereafter performance_Name , which shows the name , From_date which shows the starting date , To_date which shows the last

date , Time – declares the time of performances , then security check which express the security check and last one is status which shows the current Status.

Stage – This entity has 5 attributes, Stage_No. where we can see a different/unique stage number, Stage_name shows the name of stage, then capacity – shows the capacity of stage , then location – for proper location and last one is description which express full description about it.



Relational model:

- **Agent** (Agent_ID, Agent_Name, phone, Email, Agent_Role)
- **Team** (Team_ID, Agent_ID*, Name, Description, row3)
- **Members** (Member_ID, Team_ID*, Badge_No, Full_Name, Job_type, Staff_Role)

- **Team_Performance** (Performance_ID, Team_ID*, Stage_No, Performance_Name, From_date, To_date, Time, Security_Check, Status)
- **Stage** (Stage_No, Stage_Name, Capacity, Location, Description)

Cardinality:

- **Team 1: M Team**
- **Members M:1 Team**
- **Stage 1: M Team_Performance**
- **Team 1:M Team_Performance**

Q 1.2 Design Tables (marks -10)

Create Table Agent(

```
Agent_IDnumber(10),
Agent_Name Varchar2(50),
phone Varchar2(50),
Email Varchar2(50),
Agent_Role Varchar2(50),
Primary Key (Agent_ID)
);
```

Create Table Team(

```
Team_IDnumber(10),
Agent_IDnumber(10) Not Null,
Name Varchar2(50),
Description Varchar2(50),
row3 Varchar2(20),
Primary Key(Team_ID),
```

```
CONSTRAINT FK_Team_Agent_Agent_ID FOREIGN KEY (Agent_ID) REFERENCES
Agent (Agent_ID)
);
```

```
Create Table Members(
Member_IDnumber(10),
Team_IDnumber(10) Not Null,
Badge_NoVarchar2(50),
Full_Name Varchar2(50),
Job_TypeVarchar2(50),
Staff_Role Varchar2(50),
Primary Key (Member_ID),
```

```
CONSTRAINT FK_Members_Team_Team_ID FOREIGN KEY (Team_ID) REFERENCES
Team (Team_ID)
);
```

```
Create Table Stage(
Stage_Nonumber(10),
Stage_NameVarchar2(50),
Capacity Varchar2(50),
Location Varchar2(50),
Description Varchar2(100),
Primary Key (Stage_No)
);
```

```
Create Table Team_Performance(
Performance_IDnumber(10),
```

```

Team_IDnumber(10) Not Null,
Stage_Nonumber(10) Not Null,
Performance_Name Varchar2(50),
From_Date Date,
To_Date Date,
TimeVarchar2(50),
Security_CheckVarchar2(50),
Status Varchar2(50),
Primary Key (Performance_ID),
CONSTRAINT FK_Team_Performance_Team_Team_ID FOREIGN KEY (Team_ID)
REFERENCES Team (Team_ID),
CONSTRAINT FK_Team_Performance_Stage_Stage_No FOREIGN KEY (Stage_No)
REFERENCES Stage (Stage_No)
);

```

Q 1.3 Insert Data (marks -10)

```
/* Inserting Data into Table Agent */
```

```
Insert INTO Agent(Agent_ID, Agent_Name, phone, Email, Agent_Role)
Values(1,'Jason',2348594492,'Jason@hotmail.com','Admin');
```

```
Insert INTO Agent(Agent_ID, Agent_Name, phone, Email, Agent_Role)
Values(2,'William',3457682931,'William@hotmail.com','Manager');
```

```
/* Inserting Data into Table Team*/
```

```
Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3) Values(1,1,'Tom','','');
```

```
Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3) Values(2,2,'Nate','','');
```

```
Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3)
Values(3,2,'Nathan','','');
```

```
Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3)
Values(4,1,'Drake','','');
```

```
Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3)
Values(5,1,'Joseph','','');
```

```
/* Inserting Data into Table Members */
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(1,2,222","","'Head');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(2,2,234","","'Admin');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(3,4,444","","'Composer');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(4,5,456","","'Manager');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(5,5,161","","'Singer');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(6,5,662","","'Singer');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(7,6,567","","'Composer');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(8,4,444","","'Manager');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(9,4,145","","'Head');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(10,4,556","","'Head');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(11,4,566","","'Manager');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(12,3,454","","'Singer');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(13,3,541,"",'Head');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(14,3,566,"",'Head');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(15,3,555,"",'Admin');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(16,3,444,"",'Singer');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(17,1,345,"",'Head');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(18,1,343,"",'Singer');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(19,1,321,"",'Singer');
```

```
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(20,1,221,"",'Composer');
```

```
/* Inserting Data into Table Stage */
```

```
Insert INTO Stage(Stage_No,Stage_Name,Capacity,Location,Description) Values(1,'Somber','23456','Netherlands');
```

```
Insert INTO Stage(Stage_No,Stage_Name,Capacity,Location,Description) Values(2,'Actlike','3456','Greenland');
```

```
Insert INTO Stage(Stage_No,Stage_Name,Capacity,Location,Description) Values(3,'Yes','12344','Argentina');
```

```
Insert INTO Stage(Stage_No,Stage_Name,Capacity,Location,Description) Values(4,'Aricula','2332','Paris');
```

```
Insert INTO Stage(Stage_No,Stage_Name,Capacity,Location,Description) Values(5,'Timber','38484','Wales');
```

```
/* Inserting Data into Table Team_Performance */
```

```
Insert INTO Team_Performance(Performance_ID,Team_ID,Stage_No,Performance_Name,  
From_Date, To_Date, Time, Security_Check, Status) Values(1,3,5,'Uno',To_Date('2022-01-  
01', 'YYYY/MM/DD'),To_Date('2022-01-11',  
'YYYY/MM/DD'),'03:24:45','No','Completed');
```

```
Insert INTO Performance(Performance_ID,Team_ID,Stage_No,Performance_Name,  
From_Date, To_Date, Time, Security_Check, Status)Values(2,3,6,'Art',To_Date('2022-12-  
01', 'YYYY/MM/DD'),To_Date('2022-12-13',  
'YYYY/MM/DD'),'06:25:22','Yes','Completed');
```

```
Insert INTO Performance(Performance_ID,Team_ID,Stage_No,Performance_Name,  
From_Date, To_Date, Time, Security_Check, Status)Values(3,5,7,'Mars To  
Venus',To_Date('2022-02-12', 'YYYY/MM/DD'),To_Date('2022-02-22',  
'YYYY/MM/DD'),'10:23:10','Yes','Completed');
```

```
Insert INTO Performance(Performance_ID,Team_ID,Stage_No,Performance_Name,  
From_Date, To_Date, Time, Security_Check, Status)Values(4,3,1,'Climber',To_Date('2022-  
04-23', 'YYYY/MM/DD'),To_Date('2022-04-30',  
'YYYY/MM/DD'),'02:11:30','Yes','Completed');
```

```
Insert INTO Performance(Performance_ID,Team_ID,Stage_No,Performance_Name,  
From_Date, To_Date, Time, Security_Check, Status)Values(5,2,3,'Fanta',To_Date('2022-07-  
10', 'YYYY/MM/DD'),To_Date('2022-07-31',  
'YYYY/MM/DD'),'12:23:23','Yes','Completed');
```

Oracle Screenshots:

1. Query: Create Table Agent(Agent_ID number(10), Agent_Name Varchar2(50), phone Varchar2(50), Email Varchar2(50), Agent_Role Varchar2(50), Primary Key (Agent_ID));

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar on the left lists a connection to 'Blastonbury Pop Festival -vk21aas'. The 'Worksheet' tab is active in the main area, displaying the following SQL code:

```
>Create Table Agent( Agent_ID number(10),
Agent_Name Varchar2(50),
phone Varchar2(50),
Email Varchar2(50),
Agent_Role Varchar2(50),
Primary Key (Agent_ID));
```

The 'Script Output' panel at the bottom shows the message: 'Table AGENT created.' and 'Task completed in 0.044 seconds'.

2. Query: Insert INTO Agent(Agent_ID, Agent_Name, phone, Email, Agent_Role)
Values(1,'Jason',2348594492,'Jason@hotmail.com','Admin');

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar on the left lists a connection to 'Blastonbury Pop Festival -vk21aas'. The 'Worksheet' tab is active in the main area, displaying the following SQL code:

```
Insert INTO Agent(Agent_ID, Agent_Name, phone, Email, Agent_Role)
Values(1,'Jason',2348594492,'Jason@hotmail.com','Admin');
```

The 'Script Output' panel at the bottom shows the message: '1 row inserted.' and 'Task completed in 0.115 seconds'.

3. Query: Create Table Team(Team_IDnumber(10),Agent_IDnumber(10) Not Null,Name
Varchar2(50), Description Varchar2(50),row3 Varchar2(20),Primary Key(Team_ID),
CONSTRAINT FK_Team_Agent_Agent_ID FOREIGN KEY (Agent_ID)
REFERENCES Agent (Agent_ID));

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar on the left lists 'Blastonbury Pop Festival -vk21aas' under 'Tables (Filtered)'. The 'Worksheet' tab in the main area contains the following SQL code:

```
Create Table Team(
    Team_ID number(10),
    Agent_ID number(10) Not Null,
    Name Varchar2(50),
    Description Varchar2(50),
    Row3 Varchar2(20),
    Primary Key (Team_ID),
    CONSTRAINT FK_Team_Agent_Agent_ID FOREIGN KEY (Agent_ID) REFERENCES Agent (Agent_ID)
);
```

The 'Script Output' panel below the worksheet shows the message: 'Table TEAM created.' The status bar at the bottom right indicates 'Task completed in 0.047 seconds'.

4. Query: Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3) Values(1,1,'Tom','','');

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar on the left lists 'Blastonbury Pop Festival -vk21aas' under 'Tables (Filtered)'. The 'Worksheet' tab in the main area contains the following SQL code:

```
Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3) Values(1,1,'Tom','','');
```

The 'Script Output' panel below the worksheet shows the message: '1 row inserted.' The status bar at the bottom right indicates 'Task completed in 0.159 seconds'.

5. Query: Create Table Members(Member_ID number(10), Team_ID number(10) Not Varchar2(50), Primary Key (Member_ID), CONSTRAINT FK_Members_Team_Team_ID FOREIGN KEY (Team_ID) REFERENCES Team (Team_ID));

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar on the left lists 'Blastonbury Pop Festival -vk21aas'. The 'Worksheet' tab in the center contains the following SQL code:

```

Create Table Members(
    Member_ID number(10),
    Team_ID number(10) Not Null,
    Badge_No Varchar2(50),
    Full_Name Varchar2(50),
    Job_Type Varchar2(50),
    Staff_Role Varchar2(50),
    Primary Key (Member_ID),
    CONSTRAINT FK_Members_Team_ID FOREIGN KEY (Team_ID) REFERENCES Team (Team_ID)
);

```

The 'Script Output' panel at the bottom shows the message: 'Table MEMBERS created.'.

6. Query: Create Table

Stage(Stage_No number(10), Stage_Name Varchar2(50), Capacity Varchar2(50), Location Varchar2(50), Description Varchar2(100), Primary Key (Stage_No));

The screenshot shows the Oracle SQL Developer interface. The 'Connections' sidebar on the left lists 'Blastonbury Pop Festival -vk21aas'. The 'Worksheet' tab in the center contains the following SQL code:

```

Create Table Stage(
    Stage_No number(10),
    Stage_Name Varchar2(50),
    Capacity Varchar2(50),
    Location Varchar2(50),
    Description Varchar2(100),
    Primary Key (Stage_No)
);

```

The 'Script Output' panel at the bottom shows the message: 'Table STAGE created.'

7. Query: Insert INTO Stage(Stage_No, Stage_Name, Capacity, Location, Description) Values(1, 'Somber', '23456', 'Netherlands', '')

Oracle SQL Developer : Blastonbury Pop Festival -vk21aas

File Edit View Navigate Run Source Team Tools Window Help

Connections

Blastonbury Pop Festival -vk21aas

Worksheet Query Builder

```
Insert INTO Stage(Stage_No,Stage_Name,Capacity,Location,Description)
Values(1,'Somber','23456','Netherlands','');
```

Script Output X

Task completed in 0.11 seconds

1 row inserted.

8. Query: Create Table Team_Performance(Performance_IDnumber(10),
 Team_IDnumber(10) Not Null, Stage_Nonumber(10) Not Null, Performance_Name
 Varchar2(50), From_Date Date, To_Date Date, TimeVarchar2(50),
 Security_CheckVarchar2(50), Status Varchar2(50), Primary Key (Performance_ID),
 CONSTRAINT FK_Performance_Team_Team_ID FOREIGN KEY (Team_ID)
 REFERENCES Team (Team_ID), CONSTRAINT FK_Performance_Stage_Stage_No
 FOREIGN KEY (Stage_No) REFERENCES Stage (Stage_No));

Oracle SQL Developer : Blastonbury Pop Festival -vk21aas

File Edit View Navigate Run Source Team Tools Window Help

Connections

Blastonbury Pop Festival -vk21aas

Worksheet Query Builder

```
Create Table Team_Performance(
    Performance_ID number(10),
    Team_ID number(10) Not Null,
    Stage_No number(10) Not Null,
    Performance_Name Varchar2(50),
    From_Date Date,
    To_Date Date,
    Time Varchar2(50),
    Security_Check Varchar2(50),
    Status Varchar2(50),
    Primary Key (Performance_ID),
    CONSTRAINT FK_team_Performance_Team_Team_ID FOREIGN KEY (Team_ID) REFERENCES Team (Team_ID),
    CONSTRAINT FK_Team_Performance_Stage_Stage_No FOREIGN KEY (Stage_No) REFERENCES Stage (Stage_No)
);
```

Script Output X

Task completed in 0.041 seconds

Table TEAM_PERFORMANCE created.

Q 1.4 SQL VIEWS (marks- 10)

```
CREATE VIEW AgentJobs AS
SELECT Agent.Name, COUNT(Team_Performance.Team_ID) AS NoofJobs
FROM Agent INNER JOIN
Team ON Agent.Agent_ID = Team.Agent_ID INNER JOIN
Members ON Team.Team_ID = Members.Team_ID INNER JOIN
Team_Performance ON Team.Team_ID = Team_Performance.Team_ID INNER JOIN
Stage ON Team_Performance.Stage_No = Stage.Stage_No
GROUP BY Agent.Name;
```

Select * from AgentJobs;

The screenshot shows a SQL query window with the following content:

```
CREATE VIEW AgentJobs AS
SELECT Agent.Name, COUNT(Team_Performance.Team_ID) AS NoofJobs
FROM Agent INNER JOIN
Team ON Agent.Agent_ID = Team.Agent_ID INNER JOIN
Members ON Team.Team_ID = Members.Team_ID INNER JOIN
Team_Performance ON Team.Team_ID = Team_Performance.Team_ID INNER JOIN
Stage ON Team_Performance.Stage_No = Stage.Stage_No
GROUP BY Agent.Name;
```

Below the query window, there is a status bar with the following information:

Script Output x
Task completed in 0.033 seconds

View AGENTJOBS created.

```
CREATE USER 'dp15aad'@'localhost' ;
GRANT SELECT on agentjobs TO 'dp15aad'@'localhost' WITH GRANT OPTION;
FLUSH PRIVILEGES;
```

```

127
128 • CREATE USER 'dp15aad'@'localhost' ;
129 • GRANT SELECT on agentjobs TO 'dp15aad'@'localhost' WITH GRANT OPTION;
130 • FLUSH PRIVILEGES;
131
<

```

Output:

Action Output			
#	Time	Action	Message
1	15:36:34	select * from agentjobs	1 row(s) returned
2	15:36:57	CREATE USER 'dp15aad'@'localhost'	0 row(s) affected
3	15:37:01	GRANT SELECT on agentjobs TO 'dp15aad'@'localhost' WITH GRANT OPTION	0 row(s) affected
4	15:37:04	FLUSH PRIVILEGES	0 row(s) affected

Q 1.5 SQL Trigger (marks- 10)

CREATE TABLE `myband_data`.`tblcountdelete`(`countDelete` BIGINT NOT NULL);

Delete from tblcountdelete;
 Insert Into tblcountdelete (tblcountdelete) value (0);

DROP TRIGGER IF EXISTS `myband_data`.`agent_AFTER_DELETE`;

```

DELIMITER $$

USE `myband_data`$$
CREATE DEFINER='root'@'localhost' TRIGGER `agent_AFTER_DELETE` AFTER
DELETE ON `agent` FOR EACH ROW BEGIN
DECLARE old_countDelete integer;
SET @old_countDelete := (SELECT sum(countDelete) sum_countDelete FROM
tblcountdelete );
IF old_countDelete IS NULL THEN
SET old_countDelete = 0;
END IF;
UPDATE tblcountdelete SET countDelete = (old_countDelete + 1) ;
END$$
DELIMITER ;

```

select * from tblcountdelete;

```
166  
167 • select * from tblcountdelete
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	countDelete			

▶ 0

tblcountdelete 7 ×

Output

Action Output		
#	Time	Action
1	15:38:38	select *from tblcountdelete

Message
1 row(s) returned

```
Insert INTO Agent(Agent_ID,Agent_Name,phone,Email,Agnet_Role) Values(2,'michael jordan',2345666781,'michael jordan@hotmail.com','ExManager');  
Select * from tblcountdelete;
```

```
155  
156 • Select * from tblcountdelete;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
	countDelete			

▶ 0

tblcountdelete 9 ×

Output

Action Output		
#	Time	Action
1	15:51:24	Select *from tblcountdelete

```
delete from Agent where Agent_ID = 2;  
Select * from tblcountdelete;
```

Part -2 MYSQL – PHP (Marks- 30)

Q 2.1 Moving Dataset to MySQL (marks - 20)

```
/* Tables*/
```

```
Create Table Agent(  
Agent_ID int10),  
Agent_Name Varchar(50),  
phone Varchar(50),  
Email Varchar(50),  
Agent_Role Varchar(50),  
Primary Key (Agent_ID)  
);
```

```
Create Table Team(  
Team_ID int(10),  
Agent_ID int(10) Not Null,  
Name Varchar(50),  
Description Varchar(50),  
row3 Varchar(20),  
Primary Key(Team_ID),  
CONSTRAINT FK_Team_Agent_Agent_ID FOREIGN KEY (Agent_ID) REFERENCES  
Agent (Agent_ID)  
);
```

```
Create Table Members(  
    Member_ID int(10),  
    Team_ID int (10) Not Null,  
    Badge_No Varchar(50),  
    Full_Name Varchar(50),  
    Job_Type Varchar(50),  
    Staff_Role Varchar(50),  
    Primary Key (Member_ID),  
    CONSTRAINT FK_Members_Team_Team_ID FOREIGN KEY (Team_ID) REFERENCES  
    Team (Team_ID)  
);
```

```
Create Table Stage(  
    Stage_No int(10),  
    Stage_Name Varchar(50),  
    Capacity Varchar(50),  
    Location Varchar(50),  
    Description Varchar(100),  
    Primary Key (Stage_No)  
);
```

```
Create Table Team_Performance(  
    Performance_ID int(10),  
    Team_ID int(10) Not Null,  
    Stage_No int(10) Not Null,  
    Performance_Name Varchar(50),  
    From_Date Date,  
    To_Date Date,  
    Time Varchar(50),  
    Security_Check Varchar(50),  
    Status Varchar(50),
```

Primary Key (Performance_ID),

CONSTRAINT FK_Team_Performance_Team_ID FOREIGN KEY (Team_ID)
REFERENCES Team (Team_ID),

CONSTRAINT FK_Team_Performance_Stage_Stage_No FOREIGN KEY (Stage_No)
REFERENCES Stage (Stage_No)

);

/*Inserting Data */

Insert INTO Agent(Agent_ID, Agent_Name, phone, Email, Agent_Role)
Values(1,'Jason',2348594492,'Jason@hotmail.com','Admin');

Insert INTO Agent(Agent_ID, Agent_Name, phone, Email, Agent_Role)
Values(2,'William',3457682931,'William@hotmail.com','Manager');

Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3) Values(1,1,'Tom','','');

Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3) Values(2,2,'Nate','','');

Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3)
Values(3,2,'Nathan','','');

Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(1,2,222,"",'Head');

Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type,
Staff_Role) Values(2,2,234,"",'Admin');

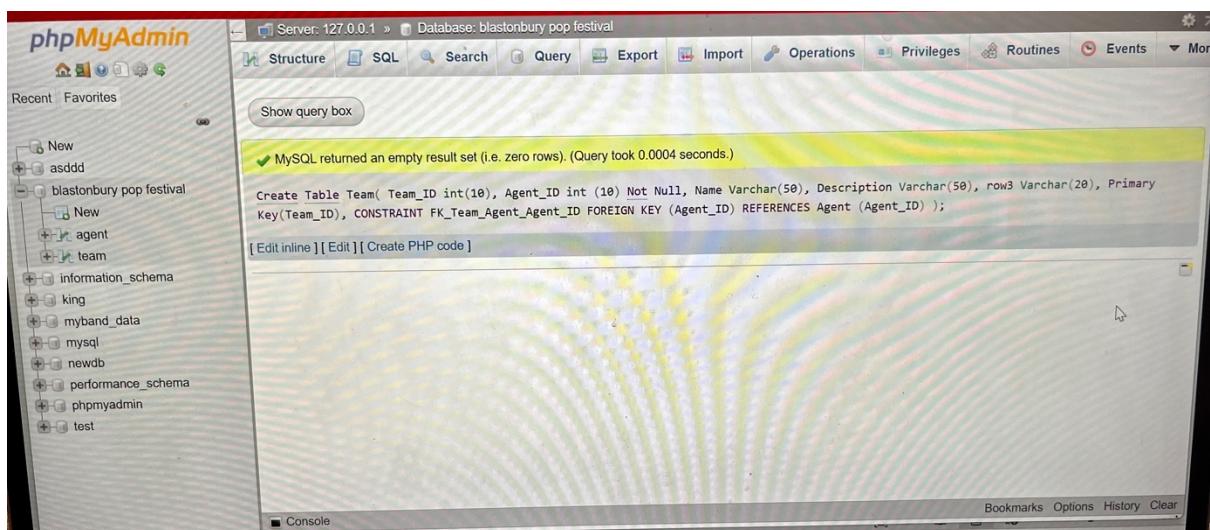
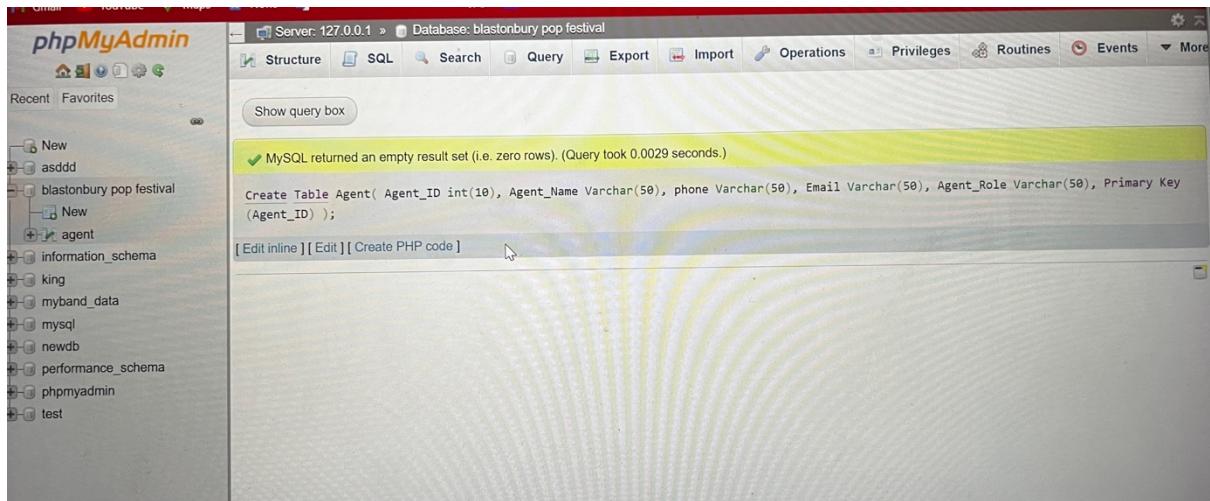
Insert INTO Stage(Stage_No,Stage_Name,Capacity,Location,Description)
Values(1,'Somber','23456','Netherlands');

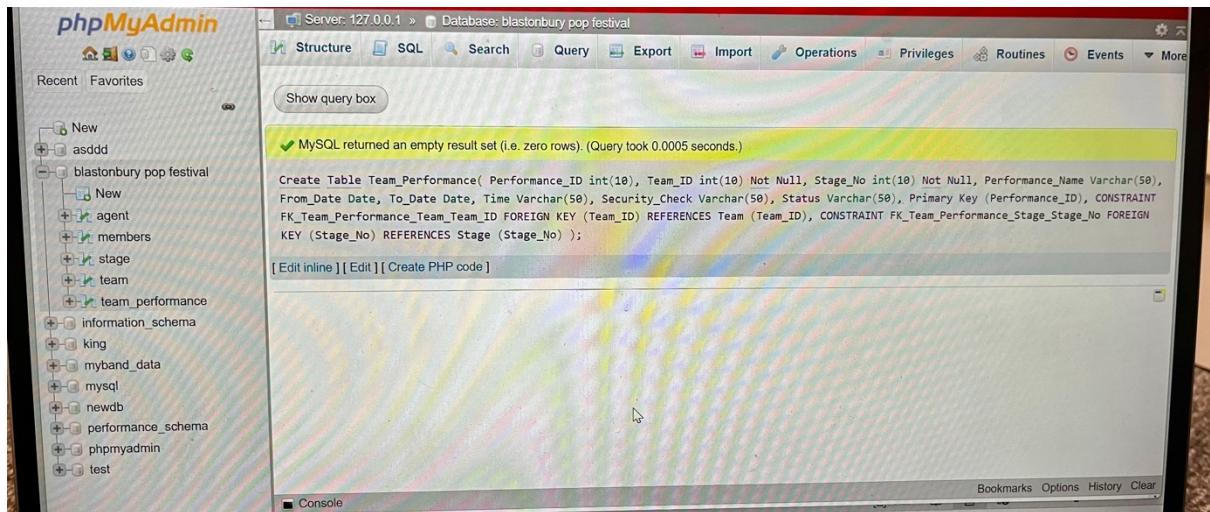
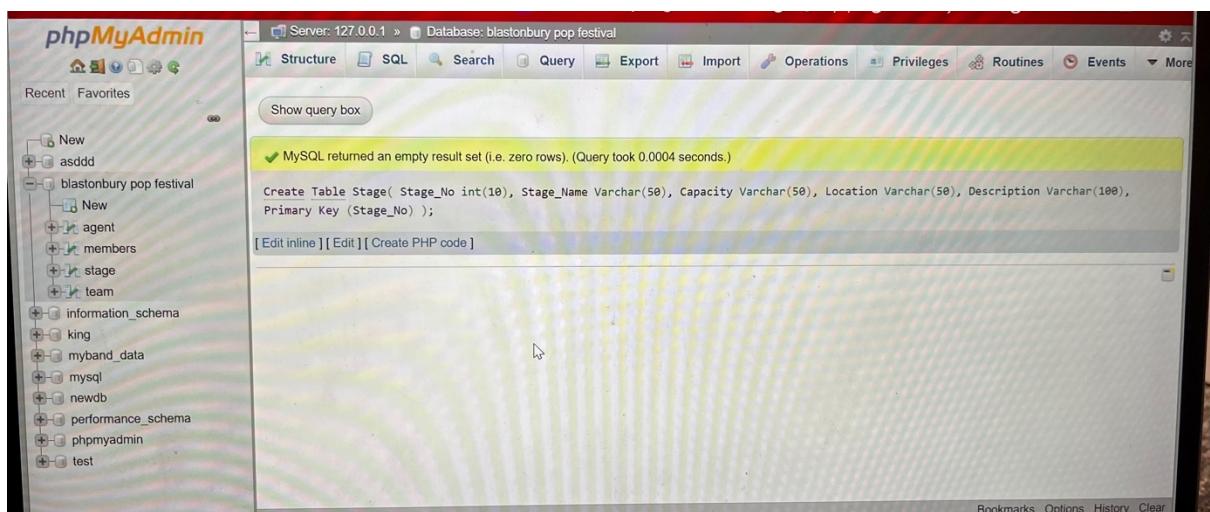
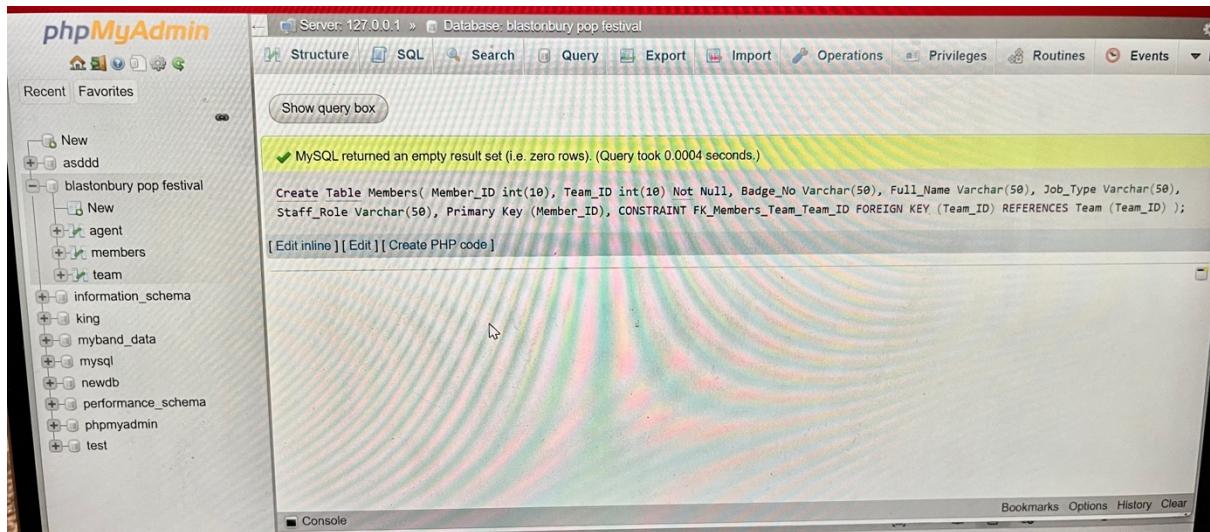
Insert INTO Stage(Stage_No,Stage_Name,Capacity,Location,Description)
Values(2,'Actlike','3456','Greenland');

Insert INTO Team_Performance(Performance_ID,Team_ID,Stage_No,Performance_Name,
From_Date, To_Date, Time, Security_Check, Status) Values(1,3,5,'Uno',To_Date('2022-01-

```
01', 'YYYY/MM/DD'),To_Date('2022-01-11',
'YYYY/MM/DD'),'03:24:45','No','Completed');
```

```
Insert INTO Performance(Performance_ID,Team_ID,Stage_No,Performance_Name,
From_Date, To_Date, Time, Security_Check, Status)Values(2,3,6,'Art',To_Date('2022-12-
01', 'YYYY/MM/DD'),To_Date('2022-12-13',
'YYYY/MM/DD'),'06:25:22','Yes','Completed');
```





phpMyAdmin

Server: 127.0.0.1 » Database: blastonbury pop festival

Structure SQL Search Query Export Import Operations Privileges Routines Events More

New asddd blastonbury pop festival New agent members stage team team_performance information_schema king myband_data mysql newdb performance_schema phpmyadmin test

Show query box

1 row inserted. (Query took 0.0004 seconds.)
Insert INTO Agent(Agent_ID, Agent_Name, phone, Email, Agent_Role) Values(1,'Jason',2348594492,'Jason@hotmail.com','Admin');

[Edit inline] [Edit] [Create PHP code]

1 row inserted. (Query took 0.0031 seconds.)
Insert INTO Agent(Agent_ID, Agent_Name, phone, Email, Agent_Role) Values(2,'William',3457682931,'William@hotmail.com','Manager');

[Edit inline] [Edit] [Create PHP code]

Console Bookmarks Options History Clear 1:28 AM

phpMyAdmin

Server: 127.0.0.1 » Database: blastonbury pop festival

Structure SQL Search Query Export Import Operations Privileges Routines Events More

New asddd blastonbury pop festival New agent members stage team team_performance information_schema king myband_data mysql newdb performance_schema phpmyadmin test

Show query box

1 row inserted. (Query took 0.0004 seconds.)
Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3) Values(1,1,'Tom','','');

[Edit inline] [Edit] [Create PHP code]

1 row inserted. (Query took 0.0019 seconds.)
Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3) Values(2,2,'Nate','','');

[Edit inline] [Edit] [Create PHP code]

1 row inserted. (Query took 0.0003 seconds.)
Insert INTO Team (Team_ID, Agent_ID, Name, Description, Row3) Values(3,2,'Nathan','','');

[Edit inline] [Edit] [Create PHP code]

Console Bookmarks Options History Clear

localhost/phpmyadmin/index.php?route=/database/sql&db=blastonbury+pop+festival

Gmail YouTube Maps News Translate vcb trans 001.jpg (15240 unread) - raj... GOOGLE.CO.IN TopUp www.holidayhomes... kvonlineadmission2...

phpMyAdmin

Server: 127.0.0.1 » Database: blastonbury pop festival

Structure SQL Search Query Export Import Operations Privileges Routines Events More

New asddd blastonbury pop festival New agent members stage team team_performance information_schema king myband_data mysql newdb performance_schema phpmyadmin test

Show query box

1 row inserted. (Query took 0.0004 seconds.)
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(1,2,222,'','Head');

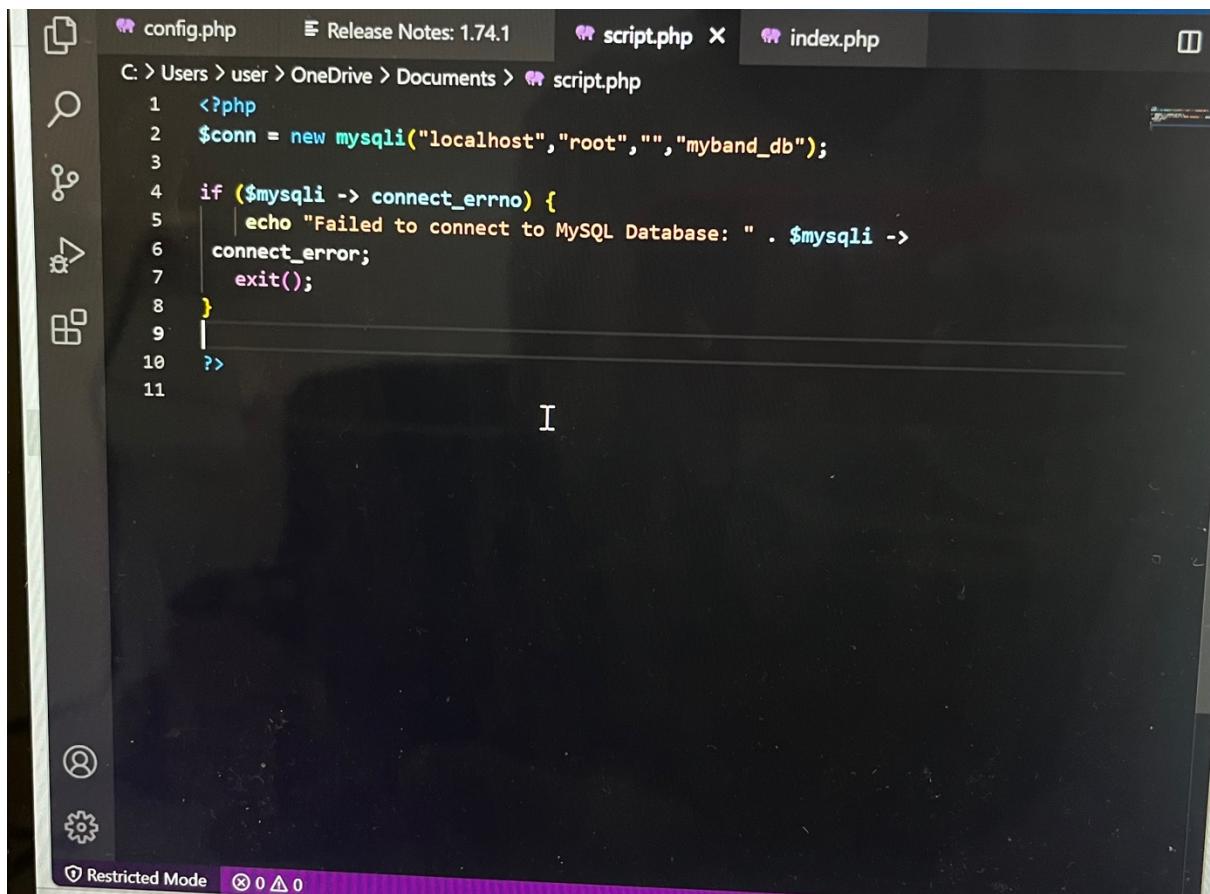
[Edit inline] [Edit] [Create PHP code]

1 row inserted. (Query took 0.0020 seconds.)
Insert INTO Members(Member_ID, Team_ID, Badge_No, Full_Name, Job_Type, Staff_Role) Values(2,2,234,'','Admin');

[Edit inline] [Edit] [Create PHP code]

Console Bookmarks Options History Clear

Q 2.2 FETCHING DATA IN PHP (marks – 10)



The screenshot shows a code editor interface with a dark theme. The top bar displays file tabs for "config.php", "Release Notes: 1.74.1", "script.php", and "index.php". The status bar at the bottom indicates "Restricted Mode" and shows statistics: 0 files, 0 errors, and 0 warnings.

```
<?php  
$conn = new mysqli("localhost","root","","myband_db");  
  
if ($conn->connect_errno) {  
    echo "Failed to connect to MySQL Database: " . $conn->  
    connect_error;  
    exit();  
}  
?  
11
```

```
C:\> Users > user > AppData > Local > Temp > Temp1_Database Copy 2[11629].zip > Database Copy 2 > index.php
 1  <?php
 2
 3  $sql = "SELECT * FROM `Agent`";
 4  $result = $conn->query($sql);
 5 ?
 6
 7  <!DOCTYPE html>
 8  <html lang="en">
 9  <head>
10  	<meta charset="UTF-8">
11  	<meta http-equiv="X-UA-Compatible" content="IE=edge">
12  	<meta name="viewport" content="width=device-width, initial-scale=1.0">
13  	<link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap@4.1.3/di
14
15  	<title>Document</title>
16  </head>
17  <body>
18
19  <div class="container mt-5">
20
21  	<table class="table">
22  	<thead class="thead-dark">
23  	<tr>
24  	<th scope="col">Agent ID</th>
25  	<th scope="col">Agent Name</th>
26  	<th scope="col">Phone</th>
27  	<th scope="col">Email </th>
28  	<th scope="col">Agent Role</th>
```

The screenshot shows a Visual Studio Code window with a dark theme. The title bar indicates the file is 'index.php' in 'Visual Studio Code'. A status bar at the bottom shows 'Ln 41 Col 51 Spaces: 4 UTF-8 CRLF PHP ⚡'. The code editor displays a PHP script that reads data from a MySQL database and prints it into an HTML table. The code includes logic to handle both successful queries (multiple rows) and empty queries (zero rows).

```
base Notes: 1.74.1 script.php index.php C:\...\Database Copy 2 index.php C:\...\Blastonbury
C: > Users > user > AppData > Local > Temp > Temp1_Database Copy 2[11629].zip > Database Copy 2 > index.php

29         </tr>
30     </thead>
31     <tbody>
32     <?php
33
34         if ($result->num_rows > 0) {
35
36             while($row = $result->fetch_assoc()) { ?>
37                 <tr>
38                     <td scope="row"><?php echo $row["Agent_ID"]; ?></td>
39                     <td><?php echo $row["Agent_Name"]; ?></td>
40                     <td><?php echo $row["Phone"]; ?></td>
41                     <td><?php echo $row["Email"]; ?></td>
42                     <td><?php echo $row["Agent_Role"]; ?></td>
43                 </tr>
44                 <?php
45             } else {
46                 echo "0 results";
47             }
48         ?>
49
50     </tbody>
51 </table>
52
53 </div>
54 </body>
55 </html>
```

Part -3 MYSQL – JAVA (Marks- 20)

Q 3.1 Using JDBC (marks- 20)

```
public class Agent {
```

```
    long Agent_ID;
    String Agent_Name;
    String phone;
```

```
String email;
String Agent_Role;

publicAgent(longAgent_ID, String Agent_Name, String phone, String email, String
Agent_Role) {

    super();
    this.Agent_ID = Agent_ID;
    this.Agent_Name = Agent_Name;
    this.phone = phone;
    this.email = email;
    this.Agent_Role = Agent_Role;
}

publicAgent() {

}

publiclonggetAgent_ID() {

    returnAgent_ID;
}

publicvoidsetAgent_ID(longAgent_ID) {

    this.Agent_ID = Agent_ID;
}

public String getAgentName() {

    returnAgent_Name;
}

publicvoidsetAgentName(String Agent_Name) {

    this.Agent_Name = Agent_Name;
}

public String getPhone() {
```

```
        return phone;
    }

public void setPhone(String phone) {
    this.phone = phone;
}

public String getEmail() {
    return email;
}

public void setEmail(String email) {
    this.email = email;
}

public String getAgentRole() {
    return Agent_Role;
}

public void setAgentRole(String Agent_Role) {
    this.Agent_Role = Agent_Role;
}

@Override

public String toString() {
    return "Agent [Agent_ID=" + Agent_ID + ", Agent_Name=" + Name + ",
phone=" + phone
            + ", email=" + email + ", Agent_Role=" + AgentRole + "]";
}

}
```

```
package simplejdbc;

public class Agent {

    long Agent_ID;
    String Agent_Name;
    String phone;
    String email;
    String Agent_Role;
    public Agent(long Agent_ID, String Agent_Name, String phone, String
email, String Agent_Role) {
        super();
        this.Agent_ID = Agent_ID;
        this.Agent_Name = Agent_Name;
        this.phone = phone;
        this.email = email;
        this.Agent_Role = Agent_Role;
    }
    public Agent() {

    }
    public long getAgent_ID() {
        return Agent_ID;
    }
    public void setAgent_ID(long Agent_ID) {
        this.Agent_ID = Agent_ID;
    }
    public String getAgent_Name() {
        return Agent_Name;
    }
    public void setAgent_Name(String Agent_Name) {
        this.Agent_Name = Agent_Name;
    }
    public String getPhone() {
        return phone;
    }
    public void setPhone(String phone) {
        this.phone = phone;
    }
    public String getEmail() {
        return email;
    }
    public void setEmail(String email) {
        this.email = email;
    }
    public String getAgent_Role() {
        return Agent_Role;
    }
    public void setAgent_Role(String Agent_Role) {
        this.Agent_Role = Agent_Role;
    }
    @Override
    public String toString() {
        return "Agent [Agent_ID=" + Agent_ID + ", Agent_Name=" + Agent_Name +
", phone=" + phone
                    + ", email=" + email + ", Agent_Role=" + Agent_Role + "]";
    }
}
```

```
package simple jdbc;
```

```
importjava.sql.Connection;
importjava.sql.DriverManager;
importjava.sql.ResultSet;
importjava.sql.Statement;
importjava.util.ArrayList;
importjava.util.List;

public class AgentJdbc {

    static Connection getConnection() {
        Connection con=null;
        try {
            Class.forName("com.mysql.jdbc.Driver");
            con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/myHibDB", "root", "root");

        }catch (Exception ex) {
            System.out.println(ex);
        }
        return con;
    }

    public static void main(String args[]) {
        List<Agent> list = new ArrayList<Agent>();
```

```

try {

    Connection con = AgentJdbc.getConnection();

    if(con!=null) {

        Statement stmt = con.createStatement();

        ResultSets = stmt.executeQuery("select * from Agent");

        while (rs.next()) {

            list.add(new Agent(rs.getLong(1), rs.getString(2),
rs.getString(3), rs.getString(4), rs.getString(5)));

        }

    }

    System.out.println("----- Please find the following Agents
List -----");

    for (AgentAgent : list) {

        System.out.println(Agent);

    }

}

else {

    System.out.println("Error while establishing DB connection ");

}

con.close();

} catch (Exception ex) {

    System.out.println(ex);

}

```

}

}