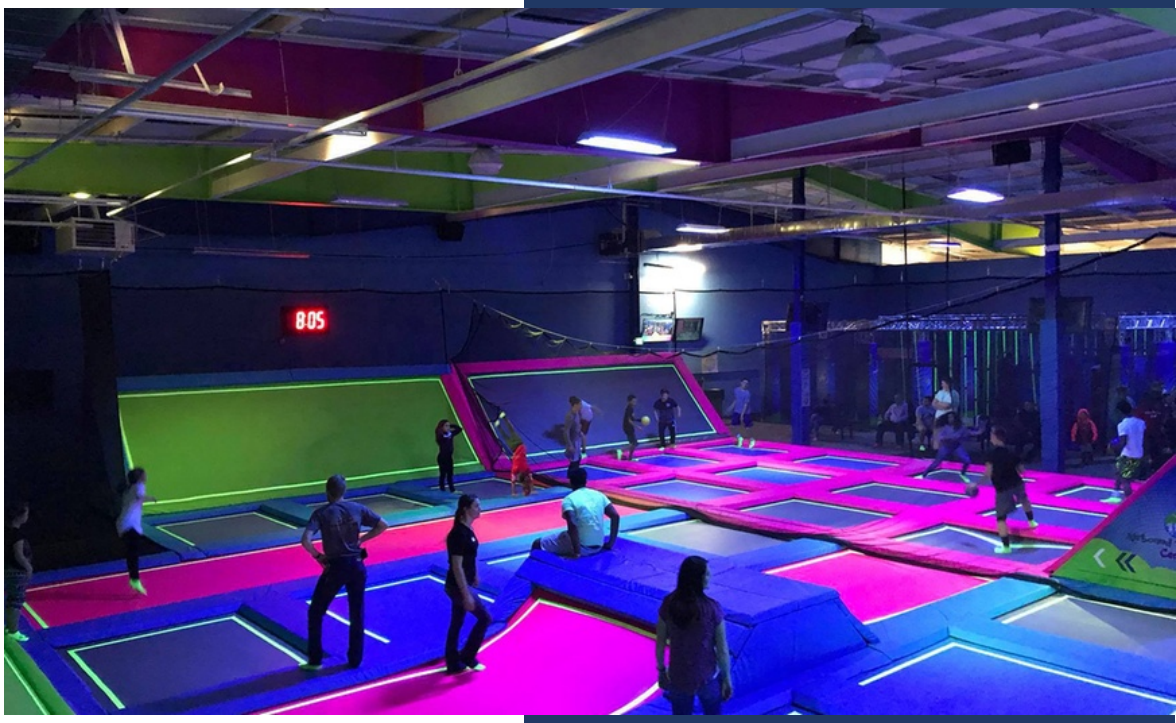
# HaNa ZONE




Data bases

Nada Hamza – 2006860

Hanadi Abdullah - 2006703

SE1

# Games Centre database Management

- **Idea**

Centre for trampoline, climbing and football. It offers monthly membership and one-day ticket.

- **Info**

in our HaNa ZONE we have several games which is trampoline, climbing and football. The customer can chose between buying a ticket or registering for packages, a ticket allow you to play only one game, when the packages allow you to play all the games in the centre for period of time the customer choose from the packages.

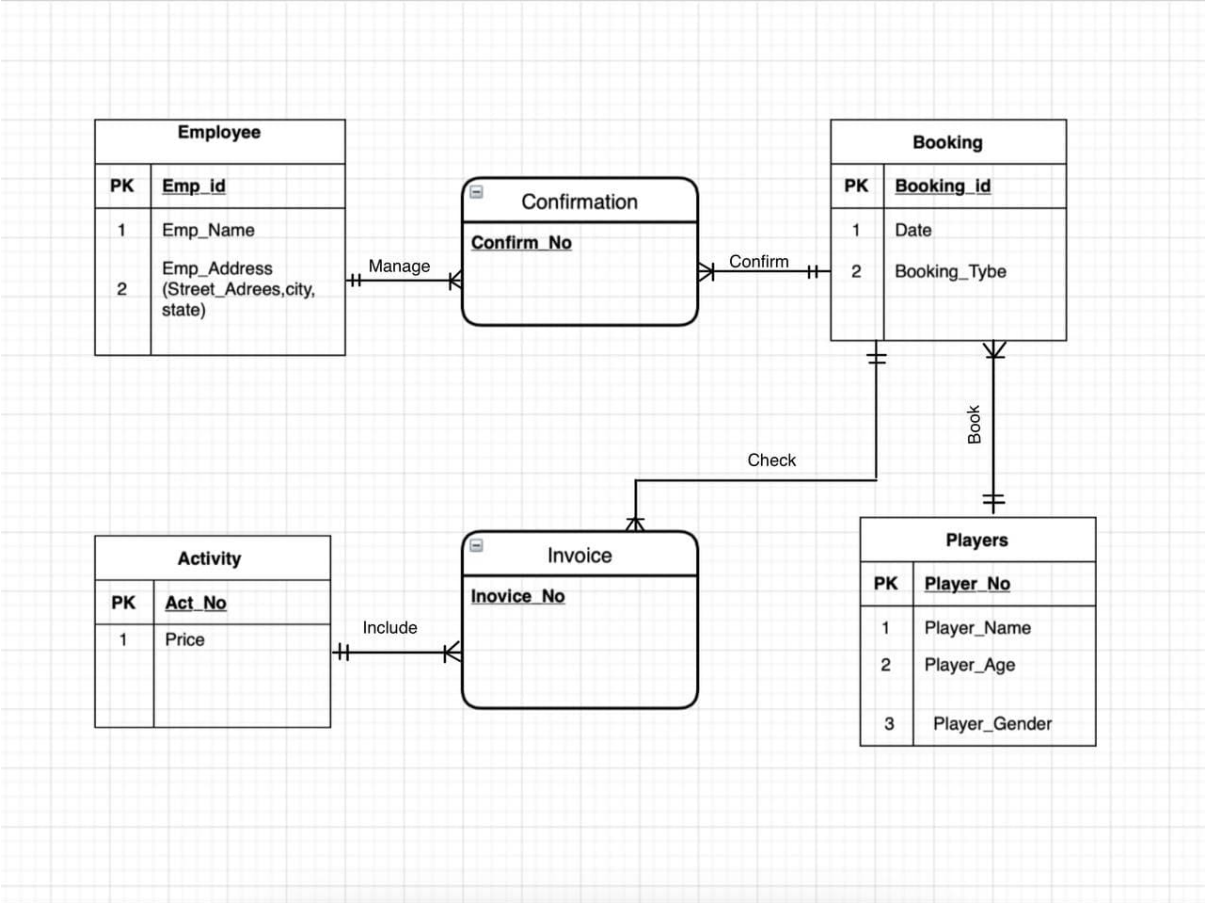
- **Entities**

1. Employee
2. Player
3. Booking
4. Activity

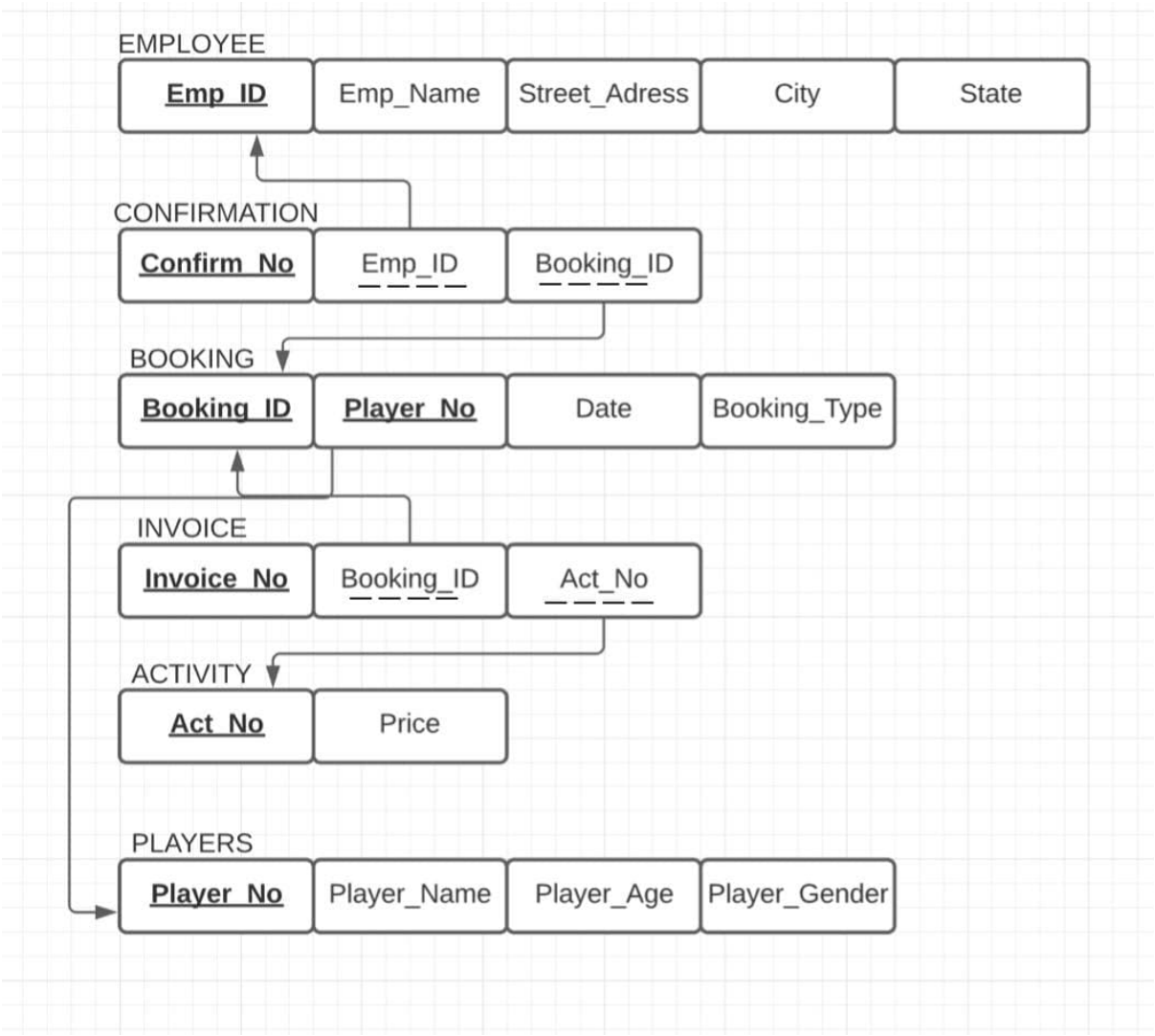
- **Business rule**

- The centre offers only three activities (trampoline, climbing, football), each activity has a unique number for it.
- Entity player has four attributes: number, name, age and gender.
- Entity booking has four attributes: ID, date, price and booking type.
- Entity employee has three attributes: ID, name and address.
- Entity activity has three attributes: activity number, booking type and price.
- A player can buy a ticket or a package but not both.
- A ticket allows you to play in one activity only, a package allows you to play in all the activities.
- An employee is responsible for booking orders for players.

- **ER Diagram**


Employee
PK
Emp_id
1
Emp_Name
2
Emp_Address (Street_Adrees,city, state)
Manage
Confirmation
Confirm_No
Confirm
Booking
PK
Booking_id
1
Date
2
Booking_Tybe
Book
Check
Activity
PK
Act_No
1
Price
Include
Invoice
Inovice_No
Players
PK
Player_No
1
Player_Name
2
Player_Age
3
Player_Gender

- **Relational Diagram**



- **Normalisation**

the relation is in **1NF** because

there are no repeating groups.

, a unique key has been identified for each relation.

And all attributes are functionally dependent on all or part of the key.

A relation is in **2NF** Because

the relation is in 1 NF

, all non-key attributes are fully functionally dependent on the

entire key (partial dependency has been removed).

the relation is in **3NF** because

the relation is in 2NF

, all transitive dependencies have been removed.

Transitive dependency: non-key attribute dependent on another non-key attribute.

- **Functional dependencies**

Employee
Emp_ID → Emp_Name , Street_Adress , City , State
Confirmation
Confirm_No → Emp_Id , Booking_ID
Booking
Booking_ID,Player_No → Date,Booking_Type
Invoice
Invoice_NO → Booking_ID,Act_No
Activity
Act_No → Price
Players
Player_No →Player_Name,Player_Age,Player_Gender

- **Physical Database implementation**

  - **Create the normalized tables.**

Live SQL

SQL Worksheet

```
1  create table employee
2  (emp_id number(6),
3  emp_name varchar2(30),
4  street_address varchar2(20),
5  city varchar2(30),
6  state varchar2(20),
7  constraint employee_pk primary key(emp_id));
```

```
Table created.
```

Live SQL

SQL Worksheet

```
1  create table players
2  (player_no number(6),
3  player_name varchar2(30),
4  player_age number(3),
5  player_gender varchar2(20),
6  constraint players_pk primary key(player_no));
```

```
Table created.
```


**SQL Worksheet**

```
1  create table activity
2  (act_no number(6),
3  price number(10),
4  constraint activity_pk primary key(act_no));
```

```
Table created.
```


**SQL Worksheet**

```
1  create table booking
2  (booking_id number(6),
3  player_no number(10),
4  booking_type varchar2(20),
5  DateE date,
6  constraint booking_pk primary key (booking_id),
7  constraint booking_fk foreign key (player_no) references players (player_no)
8  );
```

```
Table created.
```

Live SQL

SQL Worksheet

```
1  create table confirmation
2  (confirm_no number(6),
3  emp_id number(6),
4  booking_id number(6),
5  constraint confirmation_pk primary key (confirm_no));
6
```

```
Table created.
```

Live SQL

SQL Worksheet

```
1  alter table confirmation add foreign key (emp_id) references employee (emp_id);
2
```

```
Table altered.
```


**SQL Worksheet**

```
1  alter table confirmation add foreign key (booking_id) references booking (booking_id);
2
```

```
Table altered.
```


**SQL Worksheet**

```
1  create table invoice
2  (invoice_no number(6),
3  booking_id number(6),
4  act_no number(6),
5  constraint invoice_pk primary key (invoice_no),
6  constraint invoice_fk foreign key (booking_id) references booking (booking_id),
7  constraint invoiceB_fk foreign key (act_no) references activity (act_no)
8  );
```

```
Table created.
```

- **Populate your tables with 5 rows at least.**

Live SQL

SQL Worksheet

```
1  insert into employee values (123456 ,'hanadi','safa','jeddah','makkah');
2  insert into employee values  (123455 ,'nada','marwa','jeddah','makkah');
3  insert into employee values (123454 ,'lolo','samr','jeddah','makkah');
4  insert into employee values (123453 ,'waad','rabwa','jeddah','makkah');
5  insert into employee values (123452 ,'lama','bawado','jeddah','makkah');
```

```
1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.
```

Live SQL

SQL Worksheet

```
1  insert into players values (212345,'hanadi',20,'female');
2  insert into players values (212344,'abdullah',23,'male');
3  insert into players values (212343,'nada',20,'female');
4  insert into players values (212342,'khaled',19,'male');
5  insert into players values (212341,'lulu',17,'female');
```

```
1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.
```

Live SQL

SQL Worksheet

```
1  insert into  booking values (100,212345,'ticket','01-feb-21');
2  insert into  booking values (200,212344,'package','02-feb-21');
3  insert into  booking values (300,212343,'package','03-feb-21');
4  insert into  booking values (400,212342,'ticket','04-feb-21');
5  insert into  booking values (500,212341,'ticket','05-feb-21');
```

```
1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.
```

Live SQL

SQL Worksheet

```
1  insert into  confirmation values (001,123456,100);
2  insert into  confirmation values (002,123455,200);
3  insert into  confirmation values (003,123454,300);
4  insert into  confirmation values (004,123453,400);
5  insert into  confirmation values (005,123452,500);
```

```
1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.
```

Live SQL

SQL Worksheet

```
1  insert into activity values (032345,100);
2  insert into activity values (032344,200);
3  insert into activity values (032343,300);
4  insert into activity values (032342,400);
5  insert into activity values (032341,500);
```

```
1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.
```

Live SQL

SQL Worksheet

```
1  insert into  invoice values (111,100,032345);
2  insert into  invoice values (112,200,032344);
3  insert into  invoice values (113,300,032343);
4  insert into  invoice values (114,400,032342);
5  insert into  invoice values (115,500,032341);
```

```
1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.

1 row(s) inserted.
```

Live SQL

SQL Worksheet

```
1  select * from employee
```

| EMP_ID | EMP_NAME | STREET_ADDRESS | CITY | STATE |
|---|---|---|---|---|
| 123456 | hanadi | safa | jeddah | makkah |
| 123455 | nada | marwa | jeddah | makkah |
| 123454 | lolo | samr | jeddah | makkah |
| 123453 | waad | rabwa | jeddah | makkah |
| 123452 | lama | bawado | jeddah | makkah |

Download CSV

5 rows selected.

Live SQL

SQL Worksheet

```
1  select * from players
```

| PLAYER_NO | PLAYER_NAME | PLAYER_AGE | PLAYER_GENDER |
|---|---|---|---|
| 212345 | hanadi | 20 | female |
| 212344 | abdullah | 23 | male |
| 212343 | nada | 20 | female |
| 212342 | khaled | 19 | male |
| 212341 | lulu | 17 | female |

Download CSV

5 rows selected.

Live SQL

SQL Worksheet

```
1  select * from invoice
```

| INVOICE_NO | BOOKING_ID | ACT_NO |
|---|---|---|
| 111 | 100 | 32345 |
| 112 | 200 | 32344 |
| 113 | 300 | 32343 |
| 114 | 400 | 32342 |
| 115 | 500 | 32341 |

Download CSV

5 rows selected.

Live SQL

SQL Worksheet

```
1  select * from booking
```

| BOOKING_ID | PLAYER_NO | BOOKING_TYPE | DATEE |
|---|---|---|---|
| 100 | 212345 | ticket | 01-FEB-21 |
| 200 | 212344 | package | 02-FEB-21 |
| 300 | 212343 | package | 03-FEB-21 |
| 400 | 212342 | ticket | 04-FEB-21 |
| 500 | 212341 | ticket | 05-FEB-21 |

Download CSV

5 rows selected.

Live SQL

SQL Worksheet

```
1  select * from activity
```

| ACT_NO | PRICE |
|---|---|
| 32345 | 100 |
| 32344 | 200 |
| 32343 | 300 |
| 32342 | 400 |
| 32341 | 500 |

Download CSV

5 rows selected.

Live SQL

SQL Worksheet

```
1  select * from confirmation
```

| CONFIRM_NO | EMP_ID | BOOKING_ID |
|---|---|---|
| 1 | 123456 | 100 |
| 2 | 123455 | 200 |
| 3 | 123454 | 300 |
| 4 | 123453 | 400 |
| 5 | 123452 | 500 |

Download CSV

5 rows selected.

- **Design and implement at least 4 queries.**

Live SQL

SQL Worksheet

```
1  SELECT emp_name, street_address, emp_id
2  FROM employee
3  WHERE street_address = 'safa';
4
```

| EMP_NAME | STREET_ADDRESS | EMP_ID |
|---|---|---|
| hanadi | safa | 123456 |

Download CSV

Live SQL

SQL Worksheet

```
1  SELECT act_no, price
2  FROM activity
3  WHERE price >
4  (SELECT price
5  FROM activity
6  WHERE act_no= 032345);
7
```

| ACT_NO | PRICE |
|---|---|
| 32344 | 200 |
| 32343 | 300 |
| 32342 | 400 |
| 32341 | 500 |

Download CSV

4 rows selected.

Live SQL

SQL Worksheet

```
1  SELECT booking_id , player_no,dateE , booking_type
2  FROM booking
3  WHERE
4  booking_type = ANY
5  ( SELECT booking_type FROM booking
6  WHERE
7  booking_type = 'package');
```

| BOOKING_ID | PLAYER_NO | DATEE | BOOKING_TYPE |
|---|---|---|---|
| 200 | 212344 | 02-FEB-21 | package |
| 300 | 212343 | 03-FEB-21 | package |

Download CSV

2 rows selected.

Live SQL

SQL Worksheet

```
1  SELECT booking_id , player_no , datee
2  FROM booking
3  WHERE datee between TO_DATE('02/feb/21') and  TO_DATE('04/feb/21')
4  Order by datee;
```

| BOOKING_ID | PLAYER_NO | DATEE |
|---|---|---|
| 200 | 212344 | 02-FEB-21 |
| 300 | 212343 | 03-FEB-21 |
| 400 | 212342 | 04-FEB-21 |

Download CSV

3 rows selected.

- **Design two stored procedures.**


Live SQL

SQL Worksheet

```
1  Create or replace procedure new_street (ep_id employee.emp_id%Type,
2  n_street employee.street_address%Type)
3  AS
4  BEGIN
5  --Procedure to update the street address of an employee
6
7  UPDATE employee SET street_address = n_street
8  WHERE emp_id = ep_id;
9  COMMIT;
10 Dbms_output.put_line (' new street =  ' || n_street);
11 END new_street;
12
```

```
Procedure created.
```

Live SQL

SQL Worksheet

```
1  EXEC new_street (123455, 'safa');
2
```

```
Statement processed.
 new street =  safa
```

Live SQL

## SQL Worksheet

```
1  select * from employee
2
```

| EMP_ID | EMP_NAME | STREET_ADDRESS | CITY | STATE |
|---|---|---|---|---|
| 123456 | hanadi | safa | jeddah | makkah |
| 123455 | nada | safa | jeddah | makkah |
| 123454 | lolo | samr | jeddah | makkah |
| 123453 | waad | rabwa | jeddah | makkah |
| 123452 | lama | bawado | jeddah | makkah |

Download CSV

5 rows selected.