BDM FINAL GROUP PROJECT

Ayush Jain (RUID: 188002278)
Palash Nandecha (RUID: 190004525)
Akhil Patil (RUID: 190004529)
Tirth Shah (RUID: 189001132)

Part 1 (20 points)

1. Describe in at most one page (preferably less than half a page) any assumptions and constraints you made. To get full marks, in addition to any constraints mentioned in "General description" you should describe at least 2 constraints more (in addition) to the ones specified above. All key constraints should be determined in the ER below.

Assumptions:

- Every event should have one event-Id and one Id_number.

Reason – This is because every customer who is going to book the room for an event will be allocated an unique "event_id" and "id_number" which will represent details of the event type booked by the customer.

- Every room booking should have at least one id_number and one room_number.

Reason – This is because when customer books a room the room_id is marked against the customer and the customer has its own unique id_number, which represents confirmation of the booking done by the customer.

- Every event the customer opts for must have date and event_id.

Reason – Date of the event uniquely identifies on which particular date customer books the event and event_id is assigned based on the event booked.

Constraints:

Primary keys:

- CUSTOMER (ID_NUMBER), DEPENDENT (DEPENDENT_ID), RESERVATION (TRANSACTION_ID),
- ROOM(ROOM_NUMBER), EVENT(EVENT_ID).

Foreign keys:

ROOM(EVENT_ID), RESERVATION (ID_NUMBER), DEPENDENT(ID_NUMBER).

Not null:

- CUSTOMER (CUST_NAME, ID_TYPE, ID_NUM, PAYMENT_MODE)
- DEPENDENT (DEPENDENT NAME, DEPENDENT_ID)
- EVENT (EVENT_ID, EVENT_NAME, EVENT_DATE, EVENT_PRICE)
- RESERVATION (TRANSACTION_ID, BOOKING_DATE, ID_NUMBER, ROOM_NUMBER)
- ROOM (ROOM_NUMBER, ROOM_PRICE, EVENT_ID)

Explanation of additional constraints as mentioned above:

Check constraint:

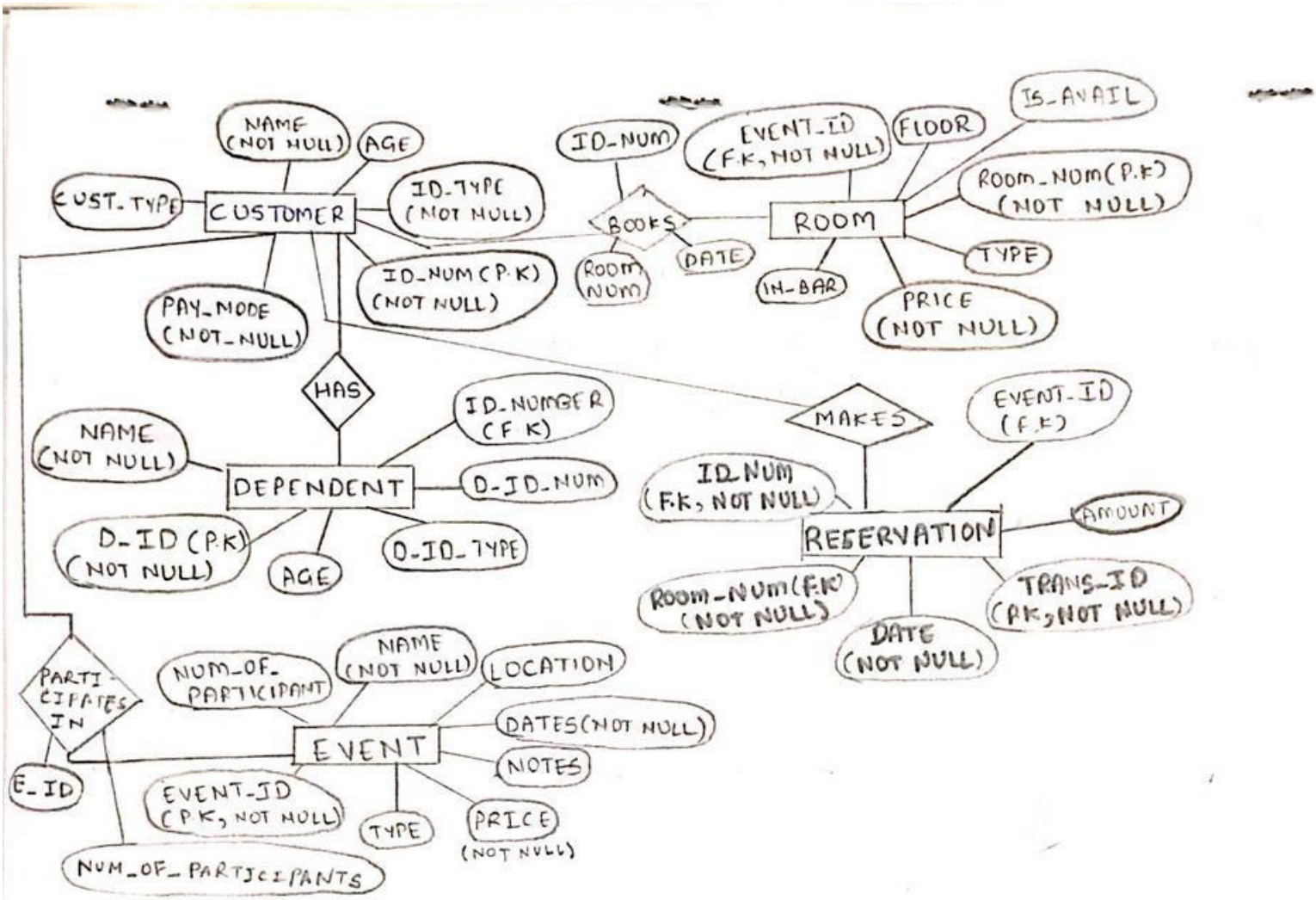
- The CHECK constraint is used to select values limited to a particular column.
- If you define a CHECK constraint on a single column it allows only certain values for this column.
- If you define a CHECK constraint on a table it can limit the values in certain columns based on values in other columns in the row.

Unique Constraint:

- The UNIQUE constraint ensures that all values in a column are different.
- Both the UNIQUE and PRIMARY KEY constraints provide a guarantee for uniqueness for a column or set of columns.
- A PRIMARY KEY constraint automatically has a UNIQUE constraint.

However, you can have many UNIQUE constraints per table, but only one PRIMARY KEY constraint per table

2. Provide a complete ER diagram that models the “General description” and includes all constraints and assumptions you wrote in (1) above.



ASSUMPTIONS :-

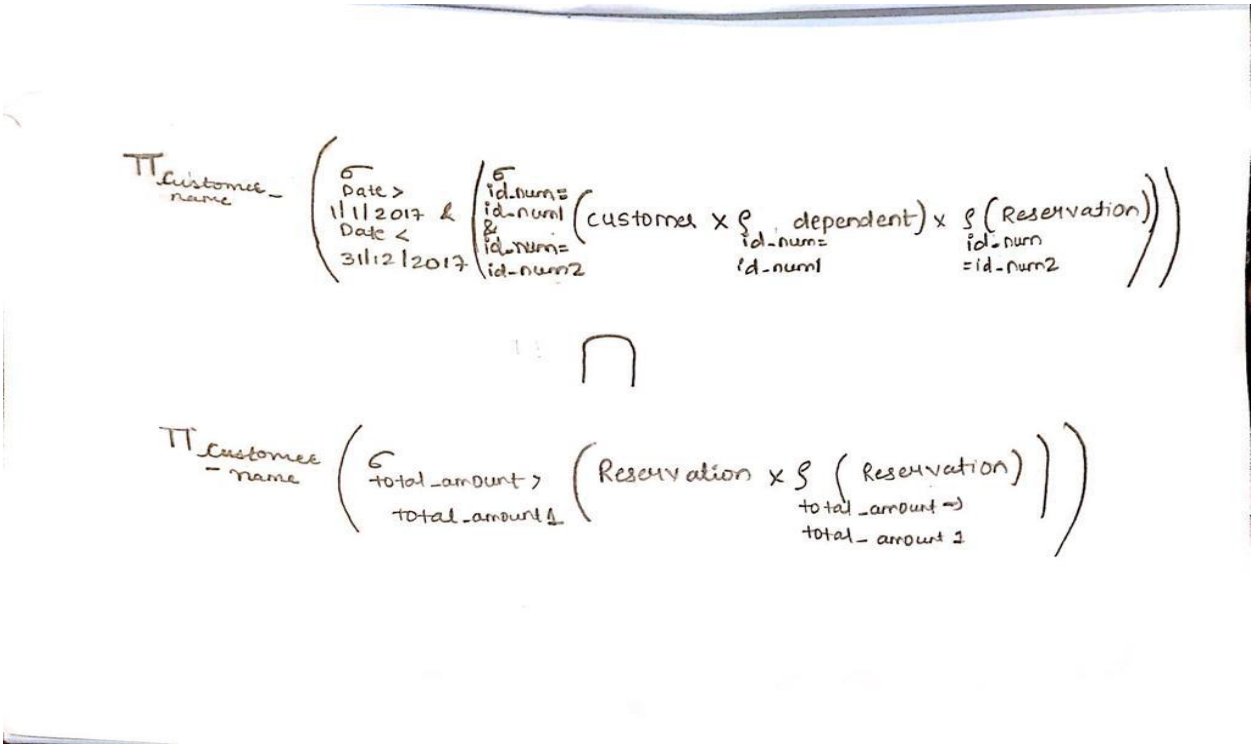
1. Every event should have one "Event_Id" and one "Id_Number".
2. Every room booking should have one "Id_Number" and one "Room_Number".
3. Every event customer opts for must have "Date" and "Event_Id".

Part 2 (20 points):

Implement the ER diagram as a Relational Model. Specify the tables you are using together with the domains (datatypes). Realize the above database and express the following query in Relational Algebra and in SQL:

Question:

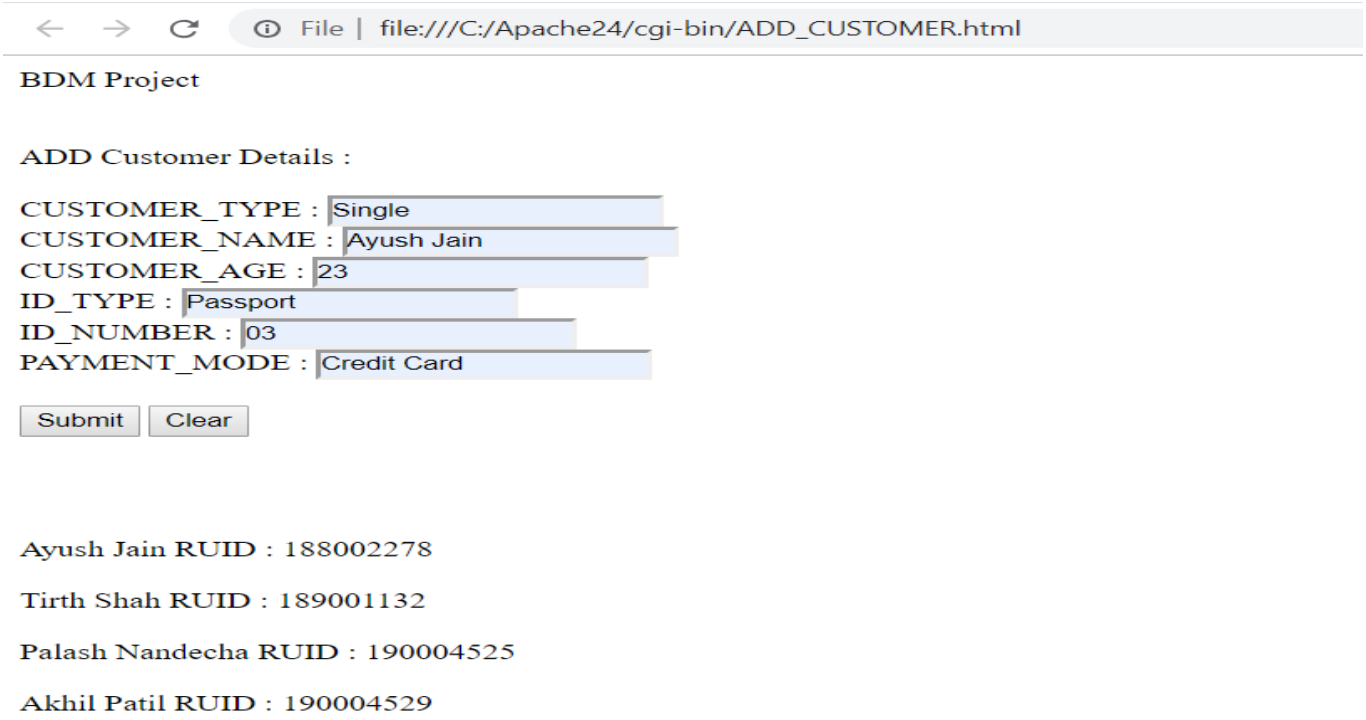
Find the customer (or customers) who paid the highest room rate in 2017 and is also related to at least one more non-primary customer.



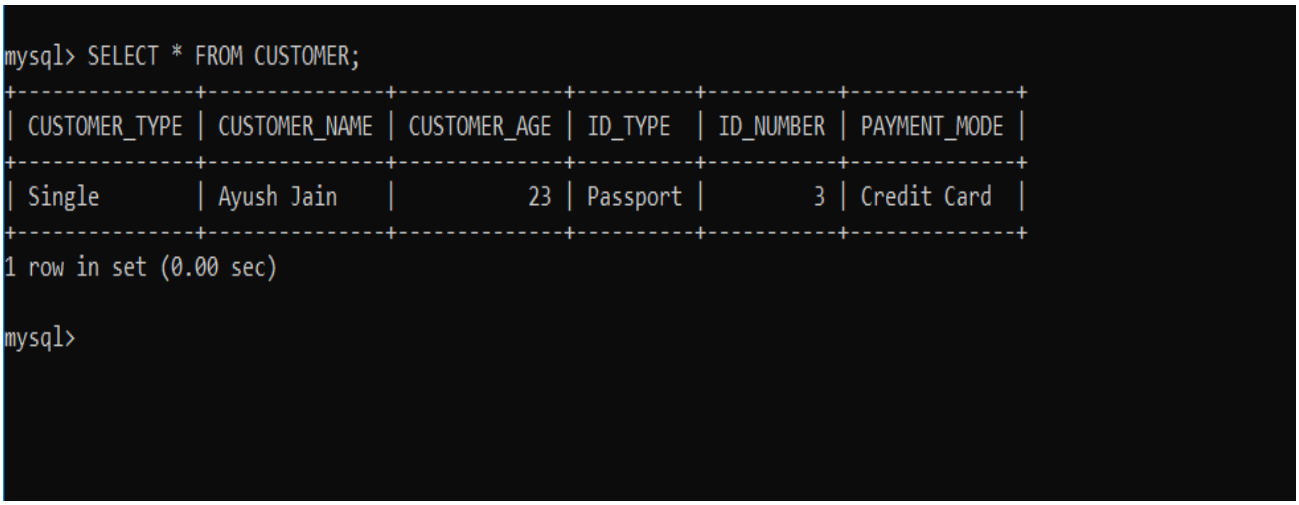
```
mysql> SELECT C.CUSTOMER_NAME , MAX(TOTAL_AMOUNT)  FROM RESERVATION R ,CUSTOMER C
    -> WHERE R.ID_NUMBER = C.ID_NUMBER AND  C.ID_NUMBER IN
    -> (SELECT C.ID_NUMBER FROM CUSTOMER C, RESERVATION R
    -> WHERE C.ID_NUMBER = R.ID_NUMBER AND YEAR(R.BOOKING_DATE) = 2017);
+---------------+-------------------+
| CUSTOMER_NAME | MAX(TOTAL_AMOUNT) |
+---------------+-------------------+
| Akhil Patil   |              5000 |
+---------------+-------------------+
1 row in set (0.00 sec)
```

Part 3 (10 points):

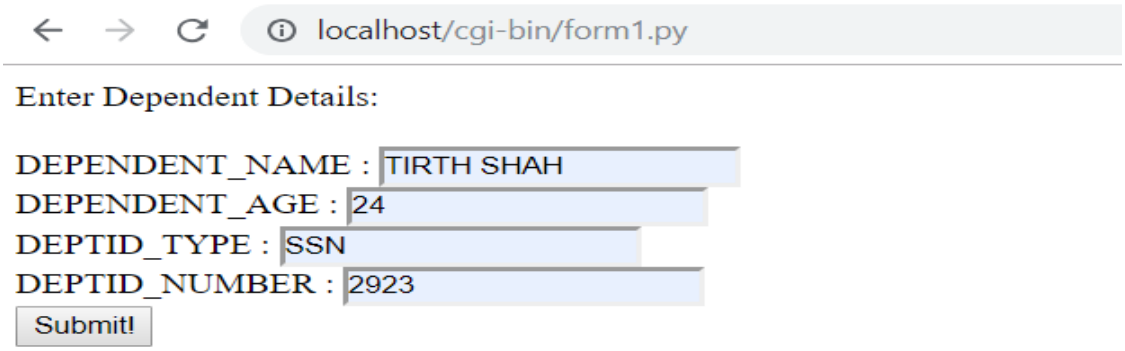
Integrate with Apache and CGI. The interface should be in HTML (very basic HTML is okay for getting full marks). For the CGI you can use Python or any other programming language you feel comfortable with.



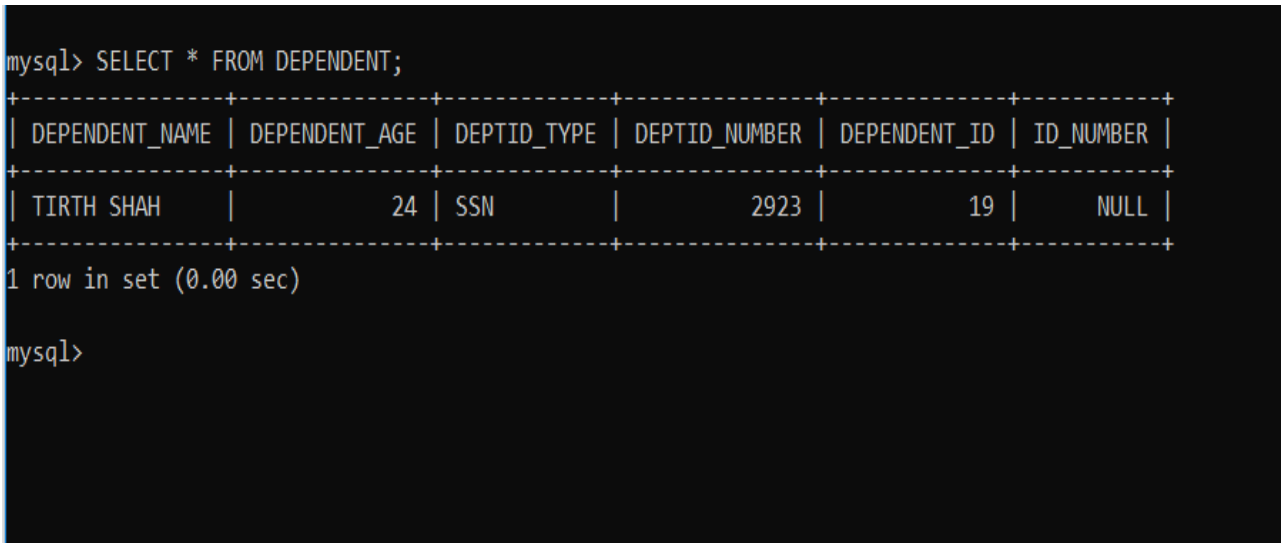
This is the Homepage where Customer Details are to be added.



As soon as details of Customers are added, database is updated by the entries.



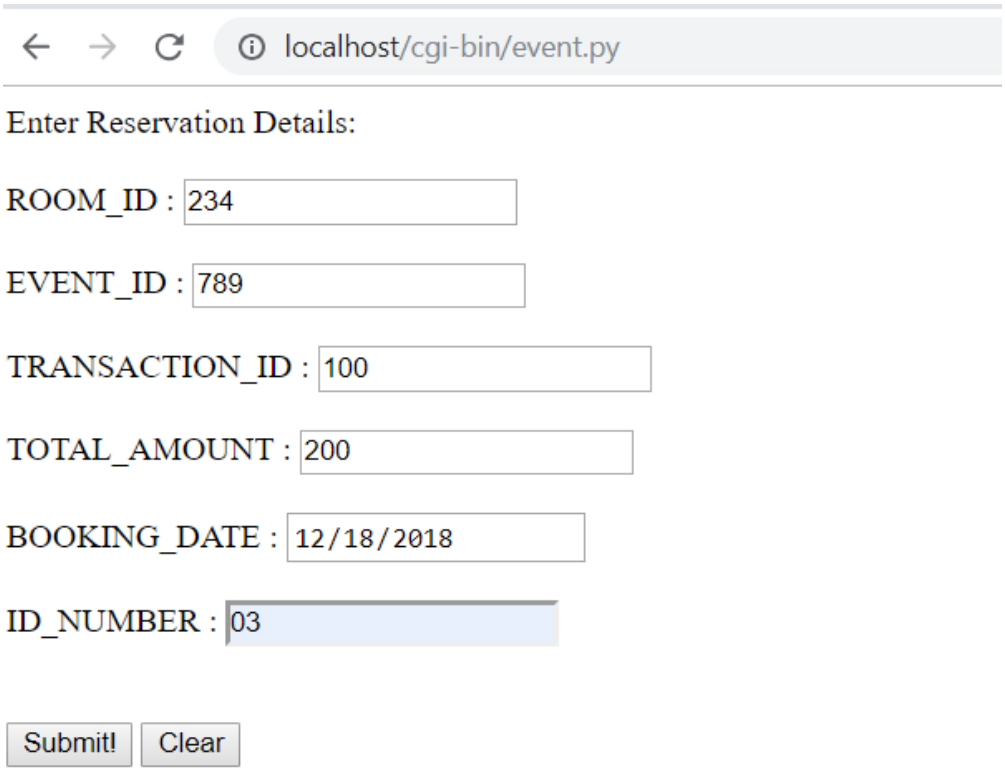
After Customer details page, next page is of Dependent where dependent details are to be added.



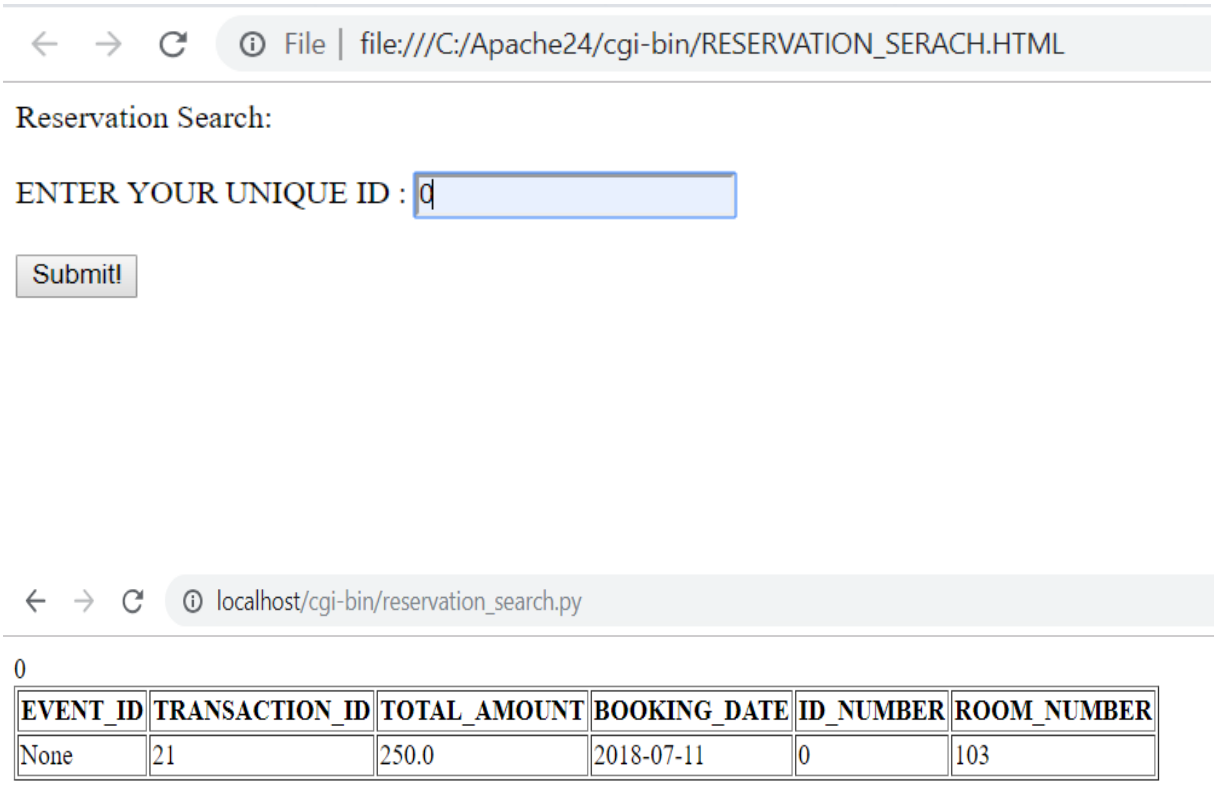
As soon as details of Dependent are added, database is updated by the entries.

Here, details of room are added and offers are applied.

Here, Event details are added.



Reservation details are then added.



EVENT_ID	TRANSACTION_ID	TOTAL_AMOUNT	BOOKING_DATE	ID_NUMBER	ROOM_NUMBER
None	21	250.0	2018-07-11	0	103

After all the details of customer and reservations are added, details of customer are retrieved using the ID_Number.

Displaying all the entries of database tables: -

```
mysql> SELECT * FROM CUSTOMER;
+---------------+-----------------+--------------+----------+-----------+--------------+
| CUSTOMER_TYPE | CUSTOMER_NAME   | CUSTOMER_AGE | ID_TYPE  | ID_NUMBER | PAYMENT_MODE |
+---------------+-----------------+--------------+----------+-----------+--------------+
| Single        | Ayush Jain      |           23 | Passport |         3 | Cash         |
| Single        | Akhil Patil     |           30 | SSN      |        20 | Cash         |
| Single        | TIRTH           |           25 | SSN      |       101 | Cash         |
| Married       | Palash Nandecha |           25 | Passport |      1000 | Credit Card  |
+---------------+-----------------+--------------+----------+-----------+--------------+
4 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM DEPENDENT;
+----------------+---------------+-------------+---------------+--------------+-----------+
| DEPENDENT_NAME | DEPENDENT_AGE | DEPTID_TYPE | DEPTID_NUMBER | DEPENDENT_ID | ID_NUMBER |
+----------------+---------------+-------------+---------------+--------------+-----------+
| TIRTH SHAH     |            24 | SSN         |          2923 |           20 |      NULL |
| ABC            |            30 | RID         |         12309 |           21 |      NULL |
| Piyush         |            30 | RID         |       1230911 |           22 |      NULL |
| Raju           |            20 | ID          |            29 |           23 |      NULL |
+----------------+---------------+-------------+---------------+--------------+-----------+
4 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM ROOM;
+-------+-------------+-----------+------------+------------+----------+-------------+
| FLOOR | ROOM_NUMBER | ROOM_TYPE | ROOM_PRICE | INROOM_BAR | EVENT_ID | isAvailable |
+-------+-------------+-----------+------------+------------+----------+-------------+
|     1 |         102 | SINGLE    |        100 | 1          |        0 |           1 |
|     1 |         103 | DOUBLE    |        250 | 0          |        0 |           0 |
|     2 |         201 | DOUBLE    |        250 | 1          |        0 |           1 |
|     2 |         202 | DELUXE    |        400 | 1          |        0 |           1 |
|     3 |         301 | EVENT     |        200 | 1          |        0 |           0 |
|     3 |         302 | SINGLE    |        100 | 0          |        0 |           1 |
+-------+-------------+-----------+------------+------------+----------+-------------+
6 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM EVENT;
+----------+---------------+----------------+--------------------+------------+------------------------+------------+-------------+
| EVENT_ID | EVENT_NAME    | EVENT_LOCATION | NUM_OF_PARTICIPANTS | EVENT_DATE | EVENT_NOTES            | EVENT_TYPE | EVENT_PRICE |
+----------+---------------+----------------+--------------------+------------+------------------------+------------+-------------+
|       11 | Family Ceremony | Newark       |                 50 | 2018-07-11 | Free Food and much more!! | Marraige   |           0 |
|       12 | Small Party   | HARRISON       |                100 | 2018-07-24 | Free Food and much more!! | Party      |           0 |
|       13 | abc           | Brunswick      |                 15 | 2018-07-20 | Optional!!             | Get to Gather |        0 |
|       14 | XYZ           | California     |                800 | 2018-07-12 | Optional!!             | Marraige   |           0 |
+----------+---------------+----------------+--------------------+------------+------------------------+------------+-------------+
4 rows in set (0.00 sec)
```

```
mysql> SELECT * FROM RESERVATION;
+----------+----------------+--------------+--------------+-----------+-------------+
| EVENT_ID | TRANSACTION_ID | TOTAL_AMOUNT | BOOKING_DATE | ID_NUMBER | ROOM_NUMBER |
+----------+----------------+--------------+--------------+-----------+-------------+
|     NULL |             20 |          200 | 2018-07-10   |        20 |         301 |
|     NULL |             21 |          250 | 2018-07-11   |        20 |         103 |
|     NULL |             22 |          250 | 2018-07-17   |         0 |         201 |
|     NULL |             23 |          400 | 2018-07-23   |         0 |         202 |
|        2 |             24 |          200 | 2017-07-03   |         0 |         203 |
|        3 |             25 |          250 | 2017-08-04   |        20 |         303 |
|        4 |             26 |          250 | 2017-07-10   |         0 |         204 |
|        5 |             27 |          100 | 2017-09-15   |         0 |         205 |
|     NULL |             28 |         5000 | 2017-09-12   |        21 |         209 |
+----------+----------------+--------------+--------------+-----------+-------------+
9 rows in set (0.00 sec)
```