A black and white sign

Description automatically generated with low confidence

**DATABASES**

October 2021 -- February 2022

**SCHOOL OF INFOCOMM TECHNOLOGY**

Diploma in Information Technology

Diploma in Data Science

Diploma in Cyber Security & Digital Forensic

**(50 % of DB Module)**

**10 Jan 2022 – 29 Jan 2022 (3 Weeks)**

|  |  |  |
| --- | --- | --- |
| Deliverables | Deadline | Submission |
| Softcopy | Saturday 29 Jan 2022, 23:59 pm | MEL Assignment Submission link |
| Hardcopy | Monday 31 Jan 2022, 10:00 am | Blk 31 Level 8 (Admin Office),  Pigeon-holes (Databases) |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| DB Class Group | | | T11 | |
| Team Number | | | 8 | Team Grade: |
| Tutor | | | Kwee Leng Chong | |
| Members | Student No. | Student Name | | Individual Grade |
| S10221788K | Lim Wee Liang Kelven | |  |
| S10222134B | Chong Yue Ming | |  |

**Penalty for late submission:**

10% of the marks will be deducted for everyday (inclusive of Saturdays, Sundays, and Public Holidays) after the deadline.

**NO** submission will be accepted after 05 Feb 2022, 23:59 pm.

[Final ER Model - 4 -](#_Toc94387259)

[List of assumptions and constraints - 4 -](#_Toc94387260)

[Assumptions: - 4 -](#_Toc94387261)

[Constraints: - 4 -](#_Toc94387262)

[List of Mapped Relations - 5 -](#_Toc94387263)

[Data Dictionary of Relations - 6 -](#_Toc94387264)

[Passenger - 6 -](#_Toc94387265)

[ContactNo - 6 -](#_Toc94387266)

[EventType - 6 -](#_Toc94387267)

[Event - 7 -](#_Toc94387268)

[EventSession - 7 -](#_Toc94387269)

[Booking - 8 -](#_Toc94387270)

[Eatery - 8 -](#_Toc94387271)

[Reservation - 9 -](#_Toc94387272)

[Cuisine - 9 -](#_Toc94387273)

[Dish - 9 -](#_Toc94387274)

[CSDish - 9 -](#_Toc94387275)

[CSOrder - 10 -](#_Toc94387276)

[FoodCategory - 10 -](#_Toc94387277)

[DishFoodCategory - 10 -](#_Toc94387278)

[Ingredient - 10 -](#_Toc94387279)

[DishIngredient - 11 -](#_Toc94387280)

[Create tables - 12 -](#_Toc94387281)

[Insertion of Tuples - 15 -](#_Toc94387282)

[Passenger - 15 -](#_Toc94387283)

[ContactNo - 15 -](#_Toc94387284)

[EventType - 16 -](#_Toc94387285)

[Event - 16 -](#_Toc94387286)

[EventSession - 17 -](#_Toc94387287)

[Booking - 18 -](#_Toc94387288)

[Eatery - 19 -](#_Toc94387289)

[Reservation - 19 -](#_Toc94387290)

[Cuisine - 20 -](#_Toc94387291)

[Dish - 20 -](#_Toc94387292)

[CSDish - 22 -](#_Toc94387293)

[CSOrder - 23 -](#_Toc94387294)

[FoodCategory - 25 -](#_Toc94387295)

[DishFoodCategory - 25 -](#_Toc94387296)

[Ingredient - 27 -](#_Toc94387297)

[DishIngredient - 28 -](#_Toc94387298)

[Display tables - 32 -](#_Toc94387299)

[Passenger - 32 -](#_Toc94387300)

[ContactNo - 32 -](#_Toc94387301)

[EventType - 33 -](#_Toc94387302)

[Event - 33 -](#_Toc94387303)

[EventSession - 33 -](#_Toc94387304)

[Booking - 34 -](#_Toc94387305)

[Eatery - 35 -](#_Toc94387306)

[Reservation - 35 -](#_Toc94387307)

[Cuisine - 35 -](#_Toc94387308)

[Dish - 36 -](#_Toc94387309)

[CSDish - 37 -](#_Toc94387310)

[CSOrder - 38 -](#_Toc94387311)

[FoodCategory - 39 -](#_Toc94387312)

[DishFoodCategory - 40 -](#_Toc94387313)

[Ingredient - 43 -](#_Toc94387314)

[DishIngredient - 45 -](#_Toc94387315)

[Queries of each member - 53 -](#_Toc94387316)

[Kelven - 53 -](#_Toc94387317)

[Yue Ming - 55 -](#_Toc94387318)

# Final ER Model

Diagram

Description automatically generated

# List of assumptions and constraints

## Assumptions:

1. Ordering of dishes are only available for cabin (room) service. Ordering of dishes in restaurants or dining halls are not included in this system.
2. No advance order of cabin service. Food will be prepared after order is placed.
3. Each dish is unique to one eatery. There is no duplication of dishes.
4. Each person attending an event (activities, entertainment, etc.) is required to have a ticket.
5. Order will be confirmed once payment is made on the app.
6. Passenger can reserve available restaurant seats and event tickets on the app.

## Constraints:

1. To avoid having some passengers from over-booking and depriving others from booking, all bookings and reservations can only be made onboard the cruise and at most 1 day in advance of required slot
2. Passenger can only use the app once they sign up for an account under their CabinNo
3. The app does not allow passengers to write reviews for the restaurants/activities/entertainments
4. The app does not allow passengers to indicate special requests (Extra ketchup, no vegetables etc)

# List of Mapped Relations

Passenger (PgrID, PgrName, PgrEmail, PgrDOB, PgrGender, CabinNo)

ContactNo (PgrContactNo, <PgrID>)

EventType (ETID, ETName)

Event (EventID, EventName, EventDescr, EventLoc, EventCapacity, EventDuration, MinAge, MaxAge, ChildPrice, AdultPrice, <ETID>)

EventSession (SessionNo, <EventID>, EventDateTime)

Booking (BookingID, BookDateTime, BookStatus, NoOfAdultTicket, NoOfChildTicket, AdultSalesPrice, ChildSalesPrice, <PgrID>, <SessionNo, <EventID>>)

Eatery (EatyID, EatyLoc, EatyName, EatyCapacity, EatyOpHr, EatyClHr)

Reservation (ReservID, ReservStatus, ReservDateTime, NoOfPax, RequiredDateTime, <PgrID>, <EatyID>)

Cuisine (cuisineID, cuisineName)

Dish (DishID, DishName, DishDescr, <cuisineID>, <EatyID>)

CSDish (<DishID>, Price)

CSOrder (<PgrID>, <DishID>, OrderDateTime, OrderPrice, OrderQty, DeliverTo, DelDateTime)

FoodCategory (FcID, FcName, FcDescr)

DishFoodCategory (<DishID>, <FcID>)

Ingredient (IngredID, IngredName)

DishIngredient (<DishID>, <IngredID>)

# Data Dictionary of Relations

## Passenger

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| PgrID | A unique key used to identify each passenger | int | Primary Key | No |
| PgrName | Name of passenger | varchar(50) |  | No |
| PgrEmail | Email of passenger | varchar(320) |  | Yes |
| PgrDOB | Date of birth of passenger | datetime |  | No |
| PgrGender | Gender of passenger | char(1) | Values: M/F | No |
| CabinNo | Cabin number of passenger | int |  | No |

## ContactNo

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| ContactNo | Contact number of passenger | int | Primary Key (ContactNo, PgrID) | No |
| PgrID | A unique key used to identify each passenger | int | Foreign Key  -> Passenger(PgrID) | No |

## EventType

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| ETID | A unique key used to identify each event type | int | Primary Key | No |
| ETName | Name of event type | varchar(30) |  | No |

## Event

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| EventID | A unique key used to identify each event | int | Primary Key | No |
| EventName | Name of event | varchar(50) |  | No |
| EventDescr | Description of event | text |  | No |
| EventLoc | Location of event | varchar(70) |  | No |
| EventCapacity | Capacity of event | int |  | No |
| EventDuration | Duration of event | time |  | Yes |
| MinAge | Minimum participation age of event | int |  | No |
| MaxAge | Maximum participation age of event | int |  | No |
| ChildPrice | Price a child must pay to participate in the event | money |  | Yes |
| AdultPrice | Price a child must pay to participate in the event | money |  | Yes |
| ETID | A unique key used to identify each event type | int | Foreign Key  -> EventType(ETID) | No |

## EventSession

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| SessionNo | A unique key that identifies the session in each event | int | Primary Key (SessionNo, EventID) | No |
| EventID | A unique key that identifies each event | int | Foreign Key  -> Event(EventID) | No |
| EventDateTime | Date and time of event | datetime |  | No |

## Booking

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| BookingID | A unique key used to identify each booking | int | Primary Key | No |
| BookingDateTime | Date and time when booking was made | datetime |  | No |
| BookStatus | Status of booking | varchar(8) | Values: Book, Confirm, WaitList | No |
| NoOfAdultTicket | Number of adult tickets required for participation in the event | int |  | No |
| NoOfChildTicket | Number of adult tickets required for participation in the event | int |  | No |
| AdultSalesPrice | The price paid by an adult to participate in the event | money |  | No |
| ChildSalesPrice | The price paid by an adult to participate in the event | money |  | No |
| PgrID | A unique key used to identify each passenger | int | Foreign Key  -> Passenger(PgrID) | No |
| SessionNo | A unique key that identifies the session in each event | int | Foreign Key  -> EventSession(SessionNo, EventID) | No |
| EventID | A unique key used to identify each event | int |  | No |

## Eatery

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| EatyID | A unique key used to identify each eatery | int | Primary Key | No |
| EatyLoc | Location of eatery | varchar(70) |  | No |
| EatyName | Name of eatery | varchar(50) |  | No |
| EatyCapacity | Capacity of eatery | int |  | No |
| EatyOpHr | Opening hour of eatery | time |  | No |
| EatyClHr | Closing hour of eatery | time |  | No |

## 

## Reservation

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| ReservID | A unique key used to identify each reservation | int | Primary Key | No |
| ReservStatus | Status of reservation | varchar(8) | Values: Book, Confirm, Waitlist | No |
| ReservDateTime | Date and time of reservation | datetime |  | No |
| NoOfPax | The number of people who will be dining in the eatery | int |  | No |
| RequiredDateTime | Date and time when passengers will dine at the eatery | datetime |  | No |
| PgrID | A unique key used to identify each passenger | int | Foreign Key  -> Passenger(PgrID) | No |
| EatyID | A unique key used to identify each eatry | int | Foreign Key  -> Eatery(EatyID) | No |

## Cuisine

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| cuisineID | A unique key used to identify each cuisine | int | Primary Key | No |
| cuisineName | Name of cuisine | varchar(50) |  | No |

## Dish

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| DishID | A unique key used to identify each dish | int | Primary Key | No |
| DishName | Name of the dish | varchar(50) |  | No |
| DishDescr | Description of the dish | text |  | No |
| cuisineID | A unique key used to identify each cuisine type | int | Foreign Key  -> Cuisine(cuisineID) | No |
| EatyID | A unique key used to identify each eatery | int | Foreign Key  -> Eatery(EatyID) | No |

## 

## CSDish

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| DishID | A unique key used to identify each dish | int | Primary Key  Foreign Key  -> Dish(DishID) | No |
| Price | The price of the cabin service dish | money |  | No |

## CSOrder

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| PgrID | A unique key used to identify each passenger | int | Foreign Key  -> Passenger(PgrID) | No |
| DishID | A unique key used to identify each dish | int | Foreign Key  -> Dish(DishID) | No |
| OrderPrice | Price paid for cabin service for dish ordered | money |  | No |
| OrderQty | Quantity of dish ordered for cabin service | int |  | No |
| OrderDateTime | Date and time cabin service was ordered | datetime | Primary Key (PgrID, DishID, OrderDateTime) | No |
| DeliverTo | CabinNo of cabin where the cabin service was ordered from | int |  | No |
| DelDateTime | Date and time when the cabin service was ordered | datetime |  | No |

## FoodCategory

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| FcID | A unique key used to identify each eatery | int | Primary Key | No |
| FcDescr | Description of food category | text |  | No |
| FcName | Name of food category | varchar(50) |  | No |

## DishFoodCategory

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| DishID | A unique key used to identify each dish | int | Primary Key (DishID, FcID)  Foreign Key  -> Dish(DishID) | No |
| FcID | A unique key used to identify each food category | int | Foreign Key  -> FoodCategory(FcID) | No |

## Ingredient

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| IngredID | A unique key used to identify each ingredient | int | Primary Key | No |
| IngredName | Name of ingredient | varchar(50) |  | No |

## DishIngredient

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Attribute** | **Description** | **Data Type** | **Constraint** | **Null Value** |
| DishID | A unique key used to identify each dish | int | Primary Key (DishID, IngredID)  Foreign Key  -> Dish(DishID) | No |
| IngredID | A unique key used to identify each ingredient | int | Foreign Key  -> Ingredient(IngredID) | No |

# Create tables

|  |
| --- |
| create database MokeCruiseBooking  use MokeCruiseBooking  create table Passenger  (  PgrID int not null Primary Key,  PgrName varchar(50) not null,  PgrEmail varchar(320) null,  PgrDOB datetime not null,  PgrGender char(1) check(PgrGender in ('M', 'F')) not null,  CabinNo int not null  )  create table ContactNo  (  PgrID int not null,  ContactNo int not null,  constraint FK\_ContactNo\_PgrID foreign key (PgrID) references Passenger(PgrID),  constraint PK\_ContactNo primary key (ContactNO, PgrID)  )  create table EventType  (  ETID int not null primary key,  ETName varchar(30) not null  )  create table Event  (  EventID int not null primary key,  EventName varchar(50) not null,  EventDescr text not null,  EventLoc varchar(70) not null,  EventCapacity int not null,  EventDuration time null,  MinAge int not null,  MaxAge int not null,  ChildPrice money null,  AdultPrice money null,  ETID int not null,  constraint FK\_Event\_ETID foreign key (ETID) references EventType(ETID)  )  create table EventSession  (  SessionNo int not null,  EventID int not null,  EventDateTime datetime not null,  constraint PK\_EventSession primary key (SessionNo, EventID),  constraint FK\_EventSession\_EventID foreign key (EventID) references Event(EventID)  )  create table Booking  (  BookingID int not null primary key,  BookingDateTime datetime not null,  BookStatus varchar(8) not null check(BookStatus in ('Book', 'Confirm', 'WaitList')),  NoOfAdultTicket int not null,  NoOfChildTicket int not null,  AdultSalesPrice money not null,  ChildSalesPrice money not null,  PgrID int not null,  SessionNo int not null,  EventID int not null,  constraint FK\_Booking foreign key (SessionNo, EventID) references EventSession(SessionNo, EventID),  constraint FK\_Booking\_PgrID foreign key (PgrID) references Passenger(PgrID)  )  create table Eatery  (  EatyID int not null primary key,  EatyLoc varchar(70) not null,  EatyName varchar(50) not null,  EatyCapacity int not null,  EatyOpHr time not null,  EatyClHr time not null  )  create table Reservation  (  ReservID int not null primary key,  BookStatus varchar(8) not null check(BookStatus in ('Book','Confirm','WaitList')),  ReservDateTime datetime not null,  NoOfPax int not null,  RequiredDateTime datetime not null,  PgrID int not null,  EatyID int not null,  constraint FK\_Reserv\_PgrID foreign key (PgrID) references Passenger(PgrID),  constraint FK\_Reserv\_EatyID foreign key (EatyID) references Eatery(EatyID)  )  create table Cuisine  (  CuisineID int not null primary key,  CuisineName varchar(50) not null  )  create table Dish  (  DishID int not null primary key,  DishName varchar(50) not null,  DishDescr text not null,  CuisineID int not null,  EatyID int not null,  constraint FK\_Dish\_CuisineID foreign key (CuisineID) references Cuisine(CuisineID),  constraint FK\_Dish\_EatyID foreign key (EatyID) references Eatery(EatyID),  )  create table CSDish  (  DishID int not null,  Price money not null,  constraint FK\_CSDish\_DishID foreign key (DishID) references Dish(DishID),  constraint PK\_CSDish primary key (DishID)  )  create table CSOrder  (  PgrID int not null,  DishID int not null,  OrderDateTime datetime not null,  OrderPrice money not null,  OrderQty int not null,  DeliverTo int not null,  DelDateTime datetime not null,  constraint FK\_Order\_PgrID foreign key (PgrID) references Passenger(PgrID),  constraint FK\_Order\_DishID foreign key (DishID) references Dish(DishID),  constraint PK\_CSOrder primary key (PgrID, DishID, OrderDateTime)  )  create table FoodCategory  (  FcID int not null primary key,  FcName varchar(50) not null,  FcDescr text not null  )  create table DishFoodCategory  (  DishID int not null,  FcID int not null,  constraint FK\_DishFoodCategory\_DishID foreign key (DishID) references Dish(DishID),  constraint FK\_DishFoodCategory\_FcID foreign key (FcID) references FoodCategory(FcID),  constraint PK\_DishFoodCategory primary key (DishID,FcID)  )  create table Ingredient  (  IngredID int not null primary key,  IngredName varchar(50) not null  )  create table DishIngredient  (  DishID int not null,  IngredID int not null,  constraint FK\_DishIngredient\_DishID foreign key (DishID) references Dish(DishID),  constraint FK\_DishIngredient\_IngredID foreign key (IngredID) references Ingredient(IngredID),  constraint PK\_DishIngredient primary key (DishID,IngredID)  ) |

# Insertion of Tuples

## Passenger

Passenger (PgrID, PgrName, PgrEmail, PgrDOB, PgrGender, CabinNo)

|  |
| --- |
| insert into Passenger values (1,'Lee Jia Jun', 'jiajun1998@gmail.com', '1998-03-07', 'M', 1001);  insert into Passenger values (2, 'Samantha Ong', 'samanthaong2@hotmail.com', '1992-04-23', 'F', 1002);  insert into Passenger values (3, 'Selena Tay', 'selenatay\_@gmail.com', '2006-01-24', 'F', 1003);  insert into Passenger values (4, 'Yong Zong Xuan', 'wanennzongxuan@gmail.com', '1973-12-03', 'M', 1004);  insert into Passenger values (5, 'Goh Wan En', 'wanennzongxuan@gmail.com', '1973-03-14', 'F', 1004);  insert into Passenger values (6, 'Muhammad Aslam Bin Syafiq', 'muhammadsyafiq@hotmail.com', '1990-05-23', 'M', 1005);  insert into Passenger values (7, 'Nur Siti Syafiq', NULL, '2018-08-30', 'F', 1005);  insert into Passenger values (8, 'Michael Tan', NULL, '2000-04-01', 'M', 1006);  insert into Passenger values (9, 'Elliot Lim', 'ellilim@hotmail.com', '2004-3-1', 'M', 1007);  insert into Passenger values (10, 'Richard Goh', 'richardgoh21@gmail.com', '1993-04-23', 'M', 1008);  insert into Passenger values (11, 'Michelle Choo', 'michooooo@gmail.com', '2002-06-06', 'F', 1009);  insert into Passenger values (12, 'Tan Jia Hao Jovan', 'jiahaotan@gmail.com', '1989-12-30', 'M', 1010);  insert into Passenger values (13, 'Tan Jia Hong', NULL, '1964-05-09', 'M', 1010); |

## ContactNo

ContactNo (<PgrID>, PgrContactNo,)

|  |
| --- |
| insert into ContactNo values (1, '67283408');  insert into ContactNo values (2, '92258667');  insert into ContactNo values (3, '62819919');  insert into ContactNo values (4, '94816079');  insert into ContactNo values (4, '62273923');  insert into ContactNo values (5, '98623388');  insert into ContactNo values (5, '87428393');  insert into ContactNo values (6, '94303773');  insert into ContactNo values (7, '94303773');  insert into ContactNo values (8, '64821463');  insert into ContactNo values (9, '67596880');  insert into ContactNo values (10, '68426177');  insert into ContactNo values (11, '84462108');  insert into ContactNo values (12, '97389462');  insert into ContactNo values (13, '97389464');  insert into ContactNo values (13, '97389463'); |

## EventType

EventType (ETID, ETName)

|  |
| --- |
| insert into EventType values (1, 'Movie');  insert into EventType values (2, 'Drawing Class');  insert into EventType values (3, 'Play');  insert into EventType values (4, 'Musical');  insert into EventType values (5, 'Water Sports');  insert into EventType values (6, 'Non-Water Sports');  insert into EventType values (7, 'Arcade');  insert into EventType values (8, 'Exercise');  insert into EventType values (9, 'Ship Tour');  insert into EventType values (10, 'Gambling'); |

## Event

Event (EventID, EventName, EventDescr, EventLoc, EventCapacity, EventDuration, MinAge, MaxAge, ChildPrice, AdultPrice, <ETID>)

|  |
| --- |
| insert into Event values (1, 'Disney''s Encanto','Another Disney movie', '#05-22', 100, '1:49:00', 7, 60, 7.00, 10.00, 1);  insert into Event values (2, 'Drawing Class For Kids','Bring your child here to learn the basics of art', '#03-12', 50, '2:00:00', 4, 12, 7.00, null, 2);  insert into Event values (3, 'Matilda', 'A play inspired by Roald Dahl''s Matilda', '#05-02', 100, '1:30:00', 7, 60, 8.00, 11.00, 3);  insert into Event values (4, 'Hamilton', 'A musical about America''s founding father', '#05-02', 100, '2:30:00', 10, 65, 7.50, 11.00, 4);  insert into Event values (5, 'Water Park', 'Enjoy the slides and pool', '#12-01', 150, '1:30:00', 7, 60, 10.00, 12.00, 5);  insert into Event values (6, 'Laser Tag', 'Have a blast playing with your family and friends at the laser tag', '#04-04', 30, '2:00:00', 7, 55, 10.00, 12.00, 6);  insert into Event values (7, 'Timezone', 'Try your luck and get jackpots', '#03-03', 50, null, 7, 60, 0, 0, 7);  insert into Event values (8, 'Spiderman: No Way Home', 'The third instalment in Marvel''s Spiderman', '#05-22', 100, '2:28:00', 7, 60, 7.00, 10.00, 1);  insert into Event values (9, 'Draw with Arty', 'Learn art with Arty', '#05-12', 50, '2:00:00', 4, 12, 7.00, null, 2);  insert into Event values (10, 'Three Kingdoms', 'It has fighting', '#05-02', 100, '1:50:00', 7, 60, 8.00, 11.00, 3);  insert into Event values (11, 'Les Miserables', 'An adaptation of Victor Hugo''s 1862 novel of the same name', '#05-02', 100, '2:30:00', 10,65, 7.50, 11.00, 4);  insert into Event values (12, 'Surfing Station','Ride the waves in style', '#12-13', 20, '1:30:00', 12, 40, 10.00, 12.00, 5);  insert into Event values (13, 'Rockwall Climbing', 'Scale the wall and reach your goal', '#12-09', 30, '1:30:00', 12, 40, 10.00, 12.00, 6);  insert into Event values (14, 'Fat Cat Arcade', 'A great place to relax with your family', '#05-01', 50, null, 7, 60, 0, 0, 7);  insert into Event values (15, 'Yoga Classes', 'Relax yourself, relax your mind', '#05-07', 30, '2:00:0'’, 15, 60, 15.00, 25.00, 8);  insert into Event values (16, 'Guided Cruise Tour', 'Journey around the ship with our Trusted Tour Guides', '#01-08', 50, null, 7, 80, 0, 0, 9);  insert into Event values (17, 'Poker Royale', 'Compete against others in Poker for a grand prize', '#07-05', 75, '3:00:00', 21, 85, null, 35.00, 10); |

## EventSession

EventSession (SessionNo, <EventID>, EventDateTime)

|  |
| --- |
| insert into EventSession values (1, 1, '06/06/2021 14:00:00');  insert into EventSession values (2, 1, '05/06/2021 16:00:00');  insert into EventSession values (1, 2, '06/06/2021 10:00:00');  insert into EventSession values (1, 3, '04/06/2021 11:00:00');  insert into EventSession values (1, 4, '06/06/2021 13:00:00');  insert into EventSession values (1, 5, '05/06/2021 10:30:00');  insert into EventSession values (1, 6, '04/06/2021 11:00:00');  insert into EventSession values (2, 6, '04/06/2021 14:00:00');  insert into EventSession values (1, 7, '06/06/2021 10:00:00');  insert into EventSession values (2, 7, '05/06/2021 10:00:00');  insert into EventSession values (1, 8, '06/06/2021 10:30:00');  insert into EventSession values (2, 8, '05/06/2021 13:30:00');  insert into EventSession values (1, 9, '06/06/2021 10:00:00');  insert into EventSession values (1, 10, '04/06/2021 12:45:00');  insert into EventSession values (1, 11, '06/06/2021 13:00:00');  insert into EventSession values (1, 12, '05/06/2021 09:45:00');  insert into EventSession values (1, 13, '04/06/2021 11:00:00');  insert into EventSession values (2, 13, '04/06/2021 13:00:00');  insert into EventSession values (1, 14, '06/06/2021 10:00:00');  insert into EventSession values (2, 14, '05/06/2021 10:00:00');  insert into EventSession values (1, 15, '04/06/2021 10:00:00');  insert into EventSession values (2, 15, '06/06/2021 10:00:00');  insert into EventSession values (1, 16, '04/06/2021 10:00:00');  insert into EventSession values (2, 16, '04/06/2021 20:30:00');  insert into EventSession values (1, 17, '05/06/2021 21:30:00'); |

## Booking

Booking (BookingID, BookDateTime, BookStatus, NoOfAdultTicket, NoOfChildTicket, AdultSalesPrice, ChildSalesPrice, <PgrID>, <SessionNo, <EventID>>)

|  |
| --- |
| insert into Booking values (1, '06/06/2021 14:00:00', 'Confirm', 1, 0, 10.00, 7.00, 1, 1, 1);  insert into Booking values (2, '05/06/2021 16:00:00', 'Book', 1, 0, 10.00, 7.00, 2, 2, 1);  insert into Booking values (3, '06/06/2021 14:00:00', 'WaitList', 0, 1, 0, 7.00, 6, 1, 2);  insert into Booking values (4, '04/06/2021 14:00:00', 'Confirm', 1, 0, 11.00, 8.00, 3, 1, 3);  insert into Booking values (5, '06/06/2021 14:00:00', 'Book', 2, 0, 11.00, 7.50, 4, 1, 4);  insert into Booking values (6, '05/06/2021 14:00:00', 'WaitList', 2, 0, 12.00, 10.00, 5, 1, 5);  insert into Booking values (7, '04/06/2021 14:00:00', 'Confirm', 1, 0, 12.00, 10.00, 6, 1, 6);  insert into Booking values (8, '04/06/2021 14:00:00', 'WaitList', 1, 0, 12.00, 10.00, 8, 2, 6);  insert into Booking values (9, '06/06/2021 14:00:00', 'Book', 1, 0, 0, 0, 9, 1, 7);  insert into Booking values (10, '05/06/2021 14:00:00', 'Confirm', 1, 0, 0, 0, 10, 2, 7);  insert into Booking values (11, '06/06/2021 14:00:00', 'Book', 1, 0, 10.00, 7.00, 11, 1, 8);  insert into Booking values (12, '05/06/2021 14:00:00', 'Confirm', 2, 0, 10.00, 7.00, 12, 2, 8);  insert into Booking values (13, '06/06/2021 14:00:00', 'WaitList', 0, 1, 0, 7.00, 6, 1, 9);  insert into Booking values (14, '04/06/2021 14:00:00', 'Book', 2, 0, 11.00, 8.00, 13, 1, 10);  insert into Booking values (15, '06/06/2021 14:00:00', 'Confirm', 1, 0, 11.00, 7.50, 1, 1, 11);  insert into Booking values (16, '05/06/2021 14:00:00', 'WaitList', 1, 0, 12.00, 10.00, 2, 1, 12);  insert into Booking values (17, '04/06/2021 14:00:00', 'Book', 1, 0, 12.00, 10.00, 3, 1, 13);  insert into Booking values (18, '04/06/2021 14:00:00', 'Confirm', 2, 0, 12.00, 10.00, 4, 2, 13);  insert into Booking values (19, '06/06/2021 14:00:00', 'WaitList', 2, 0, 0, 0, 5, 1, 14);  insert into Booking values (20, '05/06/2021 14:00:00', 'WaitList', 2, 0, 0, 0, 6, 2, 14);  insert into Booking values (21, '04/06/2021 08:00:00', 'Confirm', 1, 0, 0, 0, 11, 1, 15);  insert into Booking values (22, '05/06/2021 22:00:00', 'Confirm', 1, 0, 0, 0, 11, 2, 15);  insert into Booking values (23, '04/06/2021 08:00:00', 'Confirm', 1, 1, 0, 0, 6, 1, 16);  insert into Booking values (24, '04/06/2021 10:00:00', 'WaitList', 1, 0, 0, 0, 8, 2, 16);  insert into Booking values (25, '05/06/2021 10:00:00', 'WaitList', 1, 0, 35.00, 0, 4, 1, 17); |

## 

## Eatery

Eatery (EatyID, EatyLoc, EatyName, EatyCapacity, EatyOpHr, EatyClHr)

|  |
| --- |
| insert into Eatery values (1, '#07-22', 'Adam''s Prata House', 45, '07:30:00', '22:00:00');  insert into Eatery values (2, '#03-12', 'L''Atelier Brasserie', 135, '07:30:00', '01:00:00');  insert into Eatery values (3, '#08-02', 'Yuan Dynasty', 135, '07:30:00', '01:00:00');  insert into Eatery values (4, '#08-03', 'Isyraf''s', 135, '07:30:00', '19:00:00');  insert into Eatery values (5, '#07-15', 'The Ocean Lagoon', 135, '07:30:00', '23:00:00');  insert into Eatery values (6, '#07-03', 'Oishii', 135, '07:30:00', '23:00:00');  insert into Eatery values (7, '#03-13', 'Alberto''s Pizzareia', 105, '07:30:00', '23:00:00');  insert into Eatery values (8, '#07-17', 'Viet Corner', 35, '07:30:00', '22:00:00');  insert into Eatery values (9, '#08-04', 'Thai Cuisine', 35, '07:30:00', '22:00:00');  insert into Eatery values (10, '#03-13', 'Tasty Tacos', 35, '07:30:00', '17:00:00'); |

## Reservation

Reservation (ReservID, ReservStatus, ReservDateTime, NoOfPax, RequiredDateTime, <PgrID>, <EatyID>)

|  |
| --- |
| insert into Reservation values(1, 'Confirm', '05/06/2021 18:00:00', 1, '06/06/2021 12:00:00', 1, 1);  insert into Reservation values(2, 'Book', '04/06/2021 20:00:00', 1, '05/06/2021 08:00:00', 2, 2);  insert into Reservation values(3, 'WaitList', '05/06/2021 15:00:00', 1, '06/06/2021 09:00:00', 3, 3);  insert into Reservation values(4, 'Book', '05/06/2021 18:00:00', 2, '06/06/2021 12:00:00', 4, 4);  insert into Reservation values(5, 'WaitList', '05/06/2021 18:00:00', 2, '06/06/2021 18:30:00', 5, 5);  insert into Reservation values(6, 'Confirm', '04/06/2021 12:34:00', 2, '04/06/2021 17:45:00', 6, 6);  insert into Reservation values(7, 'Book', '04/06/2021 10:00:00', 1, '05/06/2021 12:00:00', 8, 7);  insert into Reservation values(8, 'WaitList', '05/06/2021 18:00:00', 1, '06/06/2021 19:00:00', 8, 8);  insert into Reservation values(9, 'Book', '04/06/2021 22:00:00', 1, '06/06/2021 17:30:00', 9, 9);  insert into Reservation values(10, 'WaitList', '05/06/2021 08:00:00', 1, '05/06/2021 12:30:00', 10, 10);  insert into Reservation values(11, 'Confirm', '06/06/2021 10:24:00', 1, '06/06/2021 16:00:00', 11, 1);  insert into Reservation values(12, 'WaitList', '05/06/2021 07:00:00', 2, '06/06/2021 15:45:00', 12, 2);  insert into Reservation values(13, 'Confirm', '05/06/2021 07:00:00', 2, '06/06/2021 19:00:00', 13, 3); |

## Cuisine

Cuisine (cuisineID, cuisineName)

|  |
| --- |
| insert into Cuisine values (1, 'Chinese');  insert into Cuisine values (2, 'Western');  insert into Cuisine values (3, 'Malay');  insert into Cuisine values (4, 'Indian');  insert into Cuisine values (5, 'French');  insert into Cuisine values (6, 'Japanese');  insert into Cuisine values (7, 'Italian');  insert into Cuisine values (8, 'Thai');  insert into Cuisine values (9, 'Vietnamese');  insert into Cuisine values (10, 'Mexican'); |

## Dish

Dish (DishID, DishName, DishDescr, <cuisineID>, <EatyID>)

|  |
| --- |
| insert into Dish values (1, 'Roti Prata', 'A South-Indian flatbread made by frying stretched dough flavoured with ghee', 4, 1);  insert into Dish values (2, 'Murtabak', 'Meat stuffed pancakes. With a choice between chicken or mutton', 4, 1);  insert into Dish values (3, 'Fish & Chips', 'Fried Breaded Fish served alongside a portion of fries', 2, 5);  insert into Dish values (4, 'Century Egg', 'An egg that is traditionally made in China and other Asian countries. Made by preserving eggs in a mixture of clay, salt, wood ash and quicklime from weeks to months', 1, 3);  insert into Dish values (5, 'Baguette', 'Thin type of bread of French origin that is commonly made from basic lean dough.', 5, 2);  insert into Dish values (6, 'Chicken Chop with Black Pepper Sauce', 'Grilled marinated chicken covered in a rich, bold black pepper sauce with a side of Mashed Potatoes', 2, 5);  insert into Dish values (7, 'Nasi Lemak', 'A rice dish served with roasted nuts, egg, ikan bilis, and slices of cucumber', 3, 4);  insert into Dish values (8, 'Impossible Burger', 'A vegan patty sandwiched between two lightly toasted buns.', 2, 5);  insert into Dish values (9, 'Quiche Lorraine', 'A French tart consisting of pastry crust filled with savoury custard and pieces of cheese, meat, seafood or vegetables', 5, 2);  insert into Dish values (10, 'Roasted Duck Rice', 'Duck served with a portion of rice', 1, 3);  insert into Dish values (11, 'Char Siew Rice', 'Char Siew Meat served with a portion of rice', 1, 3);  insert into Dish values (12, 'Hainanese Chicken Rice', 'Steamed Chicken served with a portion of rice ', 1, 3);  insert into Dish values (13, 'Roasted Chicken Rice', 'Roasted Chicken served with a portion of rice', 1, 3);  insert into Dish values (14, 'Poached Pear', 'Pears slowly boiled down in water', 1, 3);  insert into Dish values (15, 'Caeser Salad', 'A salad of romaine lettuce dressed in Worcestershire sauce, anchovies, garlic, Dijon mustard, parmesan cheese and black pepper', 2, 5);  insert into Dish values (16, 'Mashed Potatoes', 'Freshly picked potatoes mashed and mixed with mayonnaise and spring onions', 2, 5);  insert into Dish values (17, 'Chicken Burger', 'A chicken patty sandwiched between two lightly toasted buns', 2, 5);  insert into Dish values (18, 'Onion Rings', 'Onions sliced into rings, dipped in a batter and deep fried', 2, 5);  insert into Dish values (19, 'Cheese Fries', 'French Fries that is served with a dipping sauce consisting of Cheese and Mayonnaise', 2, 5);  insert into Dish values (20, 'French Fries', 'Evenly cut potatoes deep fried into a crispy perfection', 2, 5);  insert into Dish values (21, 'Mee Goreng', 'A savoury sweet sauce noodle mixed with chicken, prawns, vegetables and egg', 3, 4);  insert into Dish values (22, 'Chicken Satay', 'Skewered Chicken grilled on a stick served with a variety of dipping sauces', 3, 4);  insert into Dish values (23, 'Beef Satay', 'Skewered Beef grilled on a stick served with a variety of dipping sauces ', 3, 4);  insert into Dish values (24, 'Lamb Satay', 'Skewered Lamb grilled on a stick served with a variety of dipping sauces ', 3, 4);  insert into Dish values (25, 'Nasi Impit', 'Compressed rice cube served with a variety of dipping sauces ', 3, 4);  insert into Dish values (26, 'Mutton Curry', 'Curry with Mutton bits', 4, 1);  insert into Dish values (27, 'Chicken Curry', 'Curry with Chicken bits', 4, 1);  insert into Dish values (28, 'Vegetable Curry', 'Curry with an assortment of vegetables', 4, 1);  insert into Dish values (29, 'French Toast', 'Bread dipped in egg wash and sauteed until a golden brown', 5, 2);  insert into Dish values (30, 'French Onion Soup', 'Soup with a base of meat stock and onions gratineed', 5, 2);  insert into Dish values (31, 'Japanese Style Bento Set A', 'Japanese White Rice served with Chicken Karaage, Salmon and Shredded Cabbage', 6, 6);  insert into Dish values (32, 'Japanese Style Bento Set B', 'Japanese White Rice served with Chicken Karaage, Prawn Tempura and Shredded Cabbage', 6, 6);  insert into Dish values (33, 'Japanese Style Bento Set C', 'Japanese White Rice served with Salmon, Prawn Tempura and Shredded Cabbage', 6, 6);  insert into Dish values (34, 'Japanese Curry Rice Set A', 'Japanese White Rice served with Japanese Curry, Chicken Katsu and a Croquette', 6, 6);  insert into Dish values (35, 'Japanese Curry Rice Set B', 'Japanese White Rice served with Japanese Curry, Pork Katsu and a Croquette', 6, 6);  insert into Dish values (36, 'Beef Katsudon', 'A Japanese Rice Bowl made with Tonkatsu, eggs and sauteed onions simmered in a sweet savoury sauce', 6, 6);  insert into Dish values (37, 'Pepperoni Pizza', 'Tasty beef pepperoni with cheese on a tomato base', 7, 7);  insert into Dish values (38, 'Veggie Lover', 'House-made pesto blend topped with an assortment of grilled vegetables', 7, 7);  insert into Dish values (39, 'Hawaiian Pizza', 'Pineapples and Bacon with cheese on a tomato base', 7, 7);  insert into Dish values (40, 'Burrata Salad', 'Creamy burrata, cherry tomatoes, balsamic and olive oil with a perfectly seasoned salad base', 7, 7);  insert into Dish values (41, 'Aglio Olio', 'Sauteed sliced garlic in olive oil with Spaghetti Noodles, red pepper flakes and anchovies', 7, 7);  insert into Dish values (42, 'Pad Thai', 'Thai Style fried noodles with eggs, vegetables and tofu in a sauce of tamarind, fish, dried shrimp, garlic, red chilli pepper and sugar', 8, 9);  insert into Dish values (43, 'Khao Pad', 'Thai Style fried rice. Contains soy sauce, fish sauce, lime and eggs', 8, 9);  insert into Dish values (44, '2x Beef Taco', 'Beef Taco with shredded carrots, shredded cabbage and a special house-made sauce', 10, 10);  insert into Dish values (45, '2x Chicken Taco', 'Chicken Taco with shredded carrots, shredded cabbage and a special house-made sauce', 10, 10);  insert into Dish values (46, '2x Veggie Taco', 'Artificial meat with shredded carrots, shredded cabbage and a spinach puree as a sauce', 10, 10);  insert into Dish values (47, 'Banh Mi', 'Crusty Bread Rolls smeared with pate, mayo, pork chop and shredded carrot and cabbage', 9, 8);  insert into Dish values (48, 'Beef Ball Pho', 'Thin rice noodles, brown sugar, thai basil, star anise, fish sauce and grounded meat Beef Balls', 9, 8); |

## CSDish

CSDish (<DishID>, Price)

|  |
| --- |
| insert into CSDish values (1, 1.30);  insert into CSDish values (2, 9.50);  insert into CSDish values (3, 8.80);  insert into CSDish values (4, 2.00);  insert into CSDish values (5, 2.50);  insert into CSDish values (6, 10.90);  insert into CSDish values (7, 5.50);  insert into CSDish values (8, 10.90);  insert into CSDish values (9, 4.50);  insert into CSDish values (10, 4.00);  insert into CSDish values (11, 4.00);  insert into CSDish values (12, 4.00);  insert into CSDish values (13, 4.00);  insert into CSDish values (14, 1.20);  insert into CSDish values (15, 4.90);  insert into CSDish values (16, 2.60);  insert into CSDish values (17, 5.90);  insert into CSDish values (18, 3.60);  insert into CSDish values (19, 5.50);  insert into CSDish values (20, 3.90);  insert into CSDish values (21, 5.00);  insert into CSDish values (22, 1.00);  insert into CSDish values (23, 1.00);  insert into CSDish values (24, 1.00);  insert into CSDish values (25, 0.80);  insert into CSDish values (26, 3.90);  insert into CSDish values (27, 3.90);  insert into CSDish values (28, 3.50);  insert into CSDish values (29, 1.50);  insert into CSDish values (30, 4.50);  insert into CSDish values (31, 8.90);  insert into CSDish values (32, 8.90);  insert into CSDish values (33, 8.90);  insert into CSDish values (34, 7.90);  insert into CSDish values (35, 7.90);  insert into CSDish values (36, 10.90);  insert into CSDish values (37, 15.90);  insert into CSDish values (38, 15.90);  insert into CSDish values (39, 15.90);  insert into CSDish values (40, 4.90);  insert into CSDish values (41, 8.90);  insert into CSDish values (42, 6.90);  insert into CSDish values (43, 5.70);  insert into CSDish values (44, 4.00);  insert into CSDish values (45, 4.00);  insert into CSDish values (46, 4.00);  insert into CSDish values (47, 5.50);  insert into CSDish values (48, 7.90); |

## CSOrder

CSOrder (<PgrID>, <DishID>, OrderDateTime, OrderPrice, OrderQty, DeliverTo, DelDateTime)

|  |
| --- |
| insert into CSOrder values (1, 1, '04/06/2021 08:30:00', 3.90, 3, 1001, '04/06/2021 08:55:00');  insert into CSOrder values (1, 11, '04/06/2021 13:20:00', 4.00, 1, 1001, '04/06/2021 13:39:00');  insert into CSOrder values (1, 6, '04/06/2021 18:33:00', 10.90, 1, 1001, '04/06/2021 18:49:30');  insert into CSOrder values (1, 18, '04/06/2021 18:33:00', 3.60, 1, 1001, '04/06/2021 18:49:30');  insert into CSOrder values (1, 19, '04/06/2021 18:33:00', 5.50, 1, 1001, '04/06/2021 18:49:30');  insert into CSOrder values (1, 9, '05/06/2021 08:23:00', 4.50, 1, 1001, '05/06/2021 08:38:00');  insert into CSOrder values (1, 29, '05/06/2021 12:30:00', 3.90, 3, 1001, '05/06/2021 12:57:00');  insert into CSOrder values (1, 30, '05/06/2021 12:30:00', 4.50, 1, 1001, '05/06/2021 12:57:00');  insert into CSOrder values (1, 36, '05/06/2021 16:30:00', 10.90, 1, 1001, '05/06/2021 16:40:00');  insert into CSOrder values (1, 44, '06/06/2021 11:00:00', 4.50, 1, 1001, '06/06/2021 11:40:00');  insert into CSOrder values (1, 45, '06/06/2021 11:00:00', 4.50, 1, 1001, '06/06/2021 11:40:00');  insert into CSOrder values (2, 47, '04/06/2021 11:00:00', 5.50, 1, 1002, '04/06/2021 11:20:00');  insert into CSOrder values (2, 31, '04/06/2021 20:00:00', 8.90, 1, 1002, '04/06/2021 20:42:00');  insert into CSOrder values (2, 17, '06/06/2021 09:00:00', 5.90, 1, 1002, '06/06/2021 9:20:00');  insert into CSOrder values (2, 20, '06/06/2021 09:00:00', 3.90, 1, 1002, '06/06/2021 9:20:00');  insert into CSOrder values (3, 21, '04/06/2021 11:00:00', 5.00, 1, 1003, '04/06/2021 11:10:00');  insert into CSOrder values (3, 22, '04/06/2021 11:10:00', 10.00, 10, 1003, '04/06/2021 11:22:00');  insert into CSOrder values (3, 39, '04/06/2021 19:00:00', 15.90, 1, 1003, '04/06/2021 19:52:00');  insert into CSOrder values (3, 42, '05/06/2021 18:50:00', 6.90, 1, 1003, '05/06/2021 19:13:00');  insert into CSOrder values (4, 41, '04/06/2021 12:00:00', 8.90, 1, 1004, '04/06/2021 12:20:00');  insert into CSOrder values (4, 38, '04/06/2021 12:00:00', 15.90, 1, 1004, '04/06/2021 12:20:00');  insert into CSOrder values (4, 48, '04/06/2021 17:40:00', 15.80, 2, 1004, '04/06/2021 18:30:00');  insert into CSOrder values (4, 40, '04/06/2021 17:32:00', 4.90, 1, 1004, '04/06/2021 18:03:00');  insert into CSOrder values (4, 1, '05/06/2021 09:00:00', 2.60, 2, 1004, '05/06/2021 09:20:00');  insert into CSOrder values (4, 2, '06/06/2021 08:12:00', 9.50, 1, 1004, '06/06/2021 08:33:00');  insert into CSOrder values (4, 3, '06/06/2021 08:05:00', 8.80, 1, 1004, '06/06/2021 08:37:00');  insert into CSOrder values (6, 5, '04/06/2021 10:00:00', 7.50, 3, 1005, '04/06/2021 10:12:00');  insert into CSOrder values (6, 5, '06/06/2021 08:12:00', 2.50, 1, 1005, '06/06/2021 08:23:00');  insert into CSOrder values (6, 7, '06/06/2021 08:20:00', 5.50, 1, 1005, '06/06/2021 08:43:00');  insert into CSOrder values (8, 24, '04/06/2021 09:17:00', 20.00, 20, 1006, '04/06/2021 09:33:00');  insert into CSOrder values (8, 25, '04/06/2021 09:17:00', 4.00, 5, 1006, '04/06/2021 09:33:00');  insert into CSOrder values (8, 31, '04/06/2021 13:12:00', 17.80, 2, 1006, '04/06/2021 13:45:00');  insert into CSOrder values (8, 26, '04/06/2021 21:15:00', 3.90, 1, 1006, '04/06/2021 21:43:00');  insert into CSOrder values (8, 1, '05/06/2021 11:03:00', 6.50, 5, 1006, '05/06/2021 11:45:00');  insert into CSOrder values (8, 6, '05/06/2021 15:32:00', 10.90, 1, 1006, '05/06/2021 16:00:00');  insert into CSOrder values (8, 16, '05/06/2021 15:32:00', 2.60, 1, 1006, '05/06/2021 16:00:00');  insert into CSOrder values (8, 19, '05/06/2021 15:32:00', 5.50, 1, 1006, '05/06/2021 16:00:00');  insert into CSOrder values (8, 37, '05/06/2021 21:23:00', 31.80, 2, 1006, '05/06/2021 21:56:00');  insert into CSOrder values (8, 10, '06/06/2021 12:00:00', 4.00, 1, 1006, '06/06/2021 12:17:00');  insert into CSOrder values (8, 13, '06/06/2021 12:00:00', 4.00, 1, 1006, '06/06/2021 12:17:00');  insert into CSOrder values (9, 7, '04/06/2021 09:12:00', 5.50, 1, 1007, '04/06/2021 10:00:00');  insert into CSOrder values (9, 12, '04/06/2021 09:43:00', 4.00, 1, 1007, '04/06/2021 10:16:00');  insert into CSOrder values (9, 32, '04/06/2021 18:32:00', 8.90, 1, 1007, '04/06/2021 18:52:00');  insert into CSOrder values (9, 1, '05/06/2021 10:12:00', 4.20, 4, 1007, '05/06/2021 10:40:00');  insert into CSOrder values (9, 21, '05/06/2021 17:52:00', 5.00, 1, 1007, '05/06/2021 18:23:00');  insert into CSOrder values (9, 19, '05/06/2021 18:00:00', 5.00, 1, 1007, '05/06/2021 18:22:00');  insert into CSOrder values (9, 9, '06/06/2021 09:12:00', 4.50, 1, 1007, '06/06/2021 9:30:00');  insert into CSOrder values (9, 27, '06/06/2021 09:22:00', 3.90, 1, 1007, '06/06/2021 09:35:00');  insert into CSOrder values (9, 5, '06/06/2021 09:17:00', 2.50, 1, 1007, '06/06/2021 09:43:00');  insert into CSOrder values (11, 15, '04/06/2021 09:59:00', 4.90, 1, 1009, '04/06/2021 10:20:00');  insert into CSOrder values (11, 17, '04/06/2021 10:50:00', 5.90, 1, 1009, '04/06/2021 11:44:00');  insert into CSOrder values (11, 8, '05/06/2021 18:40:00', 10.90, 1, 1009, '05/06/2021 18:56:00');  insert into CSOrder values (11, 17, '05/06/2021 18:40:00', 5.90, 1, 1009, '05/06/2021 18:56:00');  insert into CSOrder values (12, 14, '04/06/2021 07:23:00', 6.00, 5, 1010, '04/06/2021 07:54:00');  insert into CSOrder values (12, 43, '04/06/2021 11:26:00', 10.40, 2, 1010, '04/06/2021 12:04:00');  insert into CSOrder values (12, 21, '05/06/2021 14:17:00', 10.00, 2, 1010, '05/06/2021 14:47:00');  insert into CSOrder values (12, 24, '06/06/2021 09:25:00', 10.00, 10, 1010, '06/06/2021 09:54:00');  insert into CSOrder values (13, 33, '04/06/2021 14:37:00', 8.90, 1, 1010, '04/06/2021 14:54:00');  insert into CSOrder values (13, 25, '06/06/2021 08:23:00', 2.40, 03, 1010, '06/06/2021 8:48:00'); |

## FoodCategory

FoodCategory (FcID, FcName, FcDescr)

|  |
| --- |
| insert into FoodCategory values (1, 'Vegetarian', 'No Animal Products(Can Include eggs)');  insert into FoodCategory values (2, 'Halal', 'Halal Approved');  insert into FoodCategory values (3, 'Dairy Free', 'No Dairy Products');  insert into FoodCategory values (4, 'Healthier Option', 'Recommended for Balanced Diet');  insert into FoodCategory values (5, 'Michelin Star', 'Michelin Star Approval');  insert into FoodCategory values (6, 'Seafood', 'Seafood');  insert into FoodCategory values (7, 'Fast Food', 'Fast Food');  insert into FoodCategory values (8, 'Fruits and Vegetables', 'Food consisting of only Fruits and Vegetables');  insert into FoodCategory values (9, 'Confectionary', 'Sweets and Chocolates');  insert into FoodCategory values (10, 'Vegan', 'No Animal Products'); |

## DishFoodCategory

DishFoodCategory (<DishID>, <FcID>)

|  |
| --- |
| insert into DishFoodCategory values (1, 1);  insert into DishFoodCategory values (1, 2);  insert into DishFoodCategory values (1, 3);  insert into DishFoodCategory values (2, 2);  insert into DishFoodCategory values (3, 7);  insert into DishFoodCategory values (3, 2);  insert into DishFoodCategory values (4, 1);  insert into DishFoodCategory values (5, 3);  insert into DishFoodCategory values (5, 4);  insert into DishFoodCategory values (5, 10);  insert into DishFoodCategory values (5, 1);  insert into DishFoodCategory values (6, 2);  insert into DishFoodCategory values (7, 4);  insert into DishFoodCategory values (7, 5);  insert into DishFoodCategory values (7, 2);  insert into DishFoodCategory values (8, 10);  insert into DishFoodCategory values (8, 1);  insert into DishFoodCategory values (8, 7);  insert into DishFoodCategory values (8, 2);  insert into DishFoodCategory values (9, 2);  insert into DishFoodCategory values (10, 3);  insert into DishFoodCategory values (11, 3);  insert into DishFoodCategory values (12, 4);  insert into DishFoodCategory values (13, 3);  insert into DishFoodCategory values (14, 2);  insert into DishFoodCategory values (15, 10);  insert into DishFoodCategory values (15, 3);  insert into DishFoodCategory values (15, 8);  insert into DishFoodCategory values (15, 4);  insert into DishFoodCategory values (15, 2);  insert into DishFoodCategory values (16, 10);  insert into DishFoodCategory values (16, 2);  insert into DishFoodCategory values (17, 7);  insert into DishFoodCategory values (17, 2);  insert into DishFoodCategory values (18, 1);  insert into DishFoodCategory values (18, 7);  insert into DishFoodCategory values (18, 2);  insert into DishFoodCategory values (19, 10);  insert into DishFoodCategory values (19, 7);  insert into DishFoodCategory values (19, 2);  insert into DishFoodCategory values (20, 10);  insert into DishFoodCategory values (20, 7);  insert into DishFoodCategory values (20, 3);  insert into DishFoodCategory values (20, 2);  insert into DishFoodCategory values (21, 2);  insert into DishFoodCategory values (21, 5);  insert into DishFoodCategory values (22, 2);  insert into DishFoodCategory values (22, 5);  insert into DishFoodCategory values (23, 2);  insert into DishFoodCategory values (23, 5);  insert into DishFoodCategory values (23, 3);  insert into DishFoodCategory values (24, 2);  insert into DishFoodCategory values (24, 5);  insert into DishFoodCategory values (24, 3);  insert into DishFoodCategory values (25, 10);  insert into DishFoodCategory values (25, 2);  insert into DishFoodCategory values (25, 3);  insert into DishFoodCategory values (25, 5);  insert into DishFoodCategory values (26, 2);  insert into DishFoodCategory values (27, 2);  insert into DishFoodCategory values (28, 10);  insert into DishFoodCategory values (28, 2);  insert into DishFoodCategory values (29, 1);  insert into DishFoodCategory values (29, 2);  insert into DishFoodCategory values (29, 3);  insert into DishFoodCategory values (30, 2);  insert into DishFoodCategory values (31, 2);  insert into DishFoodCategory values (32, 2);  insert into DishFoodCategory values (33, 2);  insert into DishFoodCategory values (34, 2);  insert into DishFoodCategory values (34, 3);  insert into DishFoodCategory values (35, 3);  insert into DishFoodCategory values (36, 2);  insert into DishFoodCategory values (37, 2);  insert into DishFoodCategory values (38, 1);  insert into DishFoodCategory values (38, 2);  insert into DishFoodCategory values (39, 2);  insert into DishFoodCategory values (40, 1);  insert into DishFoodCategory values (40, 2);  insert into DishFoodCategory values (41, 1);  insert into DishFoodCategory values (41, 2);  insert into DishFoodCategory values (42, 2);  insert into DishFoodCategory values (43, 2);  insert into DishFoodCategory values (44, 2);  insert into DishFoodCategory values (44, 3);  insert into DishFoodCategory values (45, 2);  insert into DishFoodCategory values (45, 3);  insert into DishFoodCategory values (46, 10);  insert into DishFoodCategory values (46, 2);  insert into DishFoodCategory values (46, 3);  insert into DishFoodCategory values (47, 4);  insert into DishFoodCategory values (48, 2); |

## Ingredient

Ingredient (IngredID, IngredName)

|  |
| --- |
| insert into Ingredient values (1, 'All-purpose Flour')  insert into Ingredient values (2, 'Minced Chicken')  insert into Ingredient values (3, 'Sliced Dory Fish')  insert into Ingredient values (4, 'Potato')  insert into Ingredient values (5, 'Egg')  insert into Ingredient values (6, 'Boneless Chicken Thighs')  insert into Ingredient values (7, 'Jasmine Rice')  insert into Ingredient values (8, 'Coconut Milk')  insert into Ingredient values (9, 'Chicken Wings')  insert into Ingredient values (10, 'Dried Sardines')  insert into Ingredient values (11, 'Cucumber')  insert into Ingredient values (12, 'Brioche Buns')  insert into Ingredient values (13, 'Impossible Meat')  insert into Ingredient values (14, 'Cheese')  insert into Ingredient values (15, 'De-shelled Prawns')  insert into Ingredient values (16, 'Butter')  insert into Ingredient values (17, 'Salted peanuts')  insert into Ingredient values (18, 'Black Pepper')  insert into Ingredient values (19, 'Duck')  insert into Ingredient values (20, 'Cinnamon')  insert into Ingredient values (21, 'Fennel')  insert into Ingredient values (22, 'Star anise')  insert into Ingredient values (23, 'Cloves')  insert into Ingredient values (24, 'Boneless Pork')  insert into Ingredient values (25, 'Hoisin Sauce')  insert into Ingredient values (26, 'Soy Sauce')  insert into Ingredient values (27, 'Raw Whole Chicken')  insert into Ingredient values (28, 'Pear')  insert into Ingredient values (29, 'Romaine Lettuce')  insert into Ingredient values (30, 'Worcestershire Sauce')  insert into Ingredient values (31, 'Garlic')  insert into Ingredient values (32, 'Dijon Mustard')  insert into Ingredient values (33, 'Parmesan Cheese')  insert into Ingredient values (34, 'Black Pepper')  insert into Ingredient values (35, 'Mayonnaise')  insert into Ingredient values (36, 'Spring Onions')  insert into Ingredient values (37, 'Onions')  insert into Ingredient values (38, 'Nacho Cheese')  insert into Ingredient values (39, 'Chicken Chunks')  insert into Ingredient values (40, 'Bok Choy')  insert into Ingredient values (41, 'Flank Steak')  insert into Ingredient values (42, 'Lamb Meat')  insert into Ingredient values (43, 'Peanut Sauce')  insert into Ingredient values (44, 'Curry powder')  insert into Ingredient values (45, 'Butter')  insert into Ingredient values (46, 'Gardenia Bread')  insert into Ingredient values (47, 'Japanese Petal Rice')  insert into Ingredient values (48, 'Lemon')  insert into Ingredient values (49, 'Potato Starch')  insert into Ingredient values (50, 'Cabbage')  insert into Ingredient values (51, 'Carrot')  insert into Ingredient values (52, 'Salmon')  insert into Ingredient values (53, 'Prawn')  insert into Ingredient values (54, 'Yamamori Katsu Sauce')  insert into Ingredient values (55, 'Tomato')  insert into Ingredient values (56, 'Pepperoni')  insert into Ingredient values (57, 'Pine Nuts')  insert into Ingredient values (58, 'Fresh Basil')  insert into Ingredient values (59, 'Button Mushrooms')  insert into Ingredient values (60, 'Thyme')  insert into Ingredient values (61, 'Oregano')  insert into Ingredient values (62, 'Mozzarella')  insert into Ingredient values (63, 'Pineapple')  insert into Ingredient values (64, 'Bacon Slices')  insert into Ingredient values (65, 'Burrata')  insert into Ingredient values (66, 'Balsamic Vinegar')  insert into Ingredient values (67, 'Cherry Tomato')  insert into Ingredient values (68, 'Salt')  insert into Ingredient values (69, 'Anchovies')  insert into Ingredient values (70, 'Red pepper flakes')  insert into Ingredient values (71, 'Tofu')  insert into Ingredient values (72, 'Tamarind')  insert into Ingredient values (73, 'Dried Shrimp')  insert into Ingredient values (74, 'Sugar')  insert into Ingredient values (75, 'Fish Sauce')  insert into Ingredient values (76, 'Lime')  insert into Ingredient values (77, 'Cabbage')  insert into Ingredient values (78, 'Spinach')  insert into Ingredient values (79, 'Minced Beef')  insert into Ingredient values (80, 'Brown Sugar')  insert into Ingredient values (81, 'Breadcrumbs')  insert into Ingredient values (82, 'Pork Meat')  insert into Ingredient values (83, 'Vietnamese Baguette')  insert into Ingredient values (84, 'Rice noodle') |

## DishIngredient

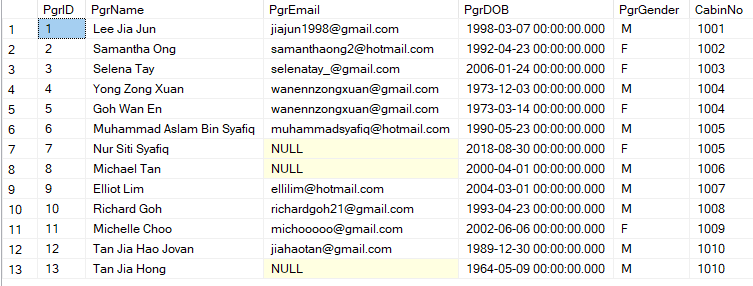
DishIngredient (<DishID>, <IngredID>)

|  |
| --- |
| insert into DishIngredient values (1, 1)  insert into DishIngredient values (1, 4)  insert into DishIngredient values (1, 5)  insert into DishIngredient values (2, 1)  insert into DishIngredient values (2, 2)  insert into DishIngredient values (2, 5)  insert into DishIngredient values (2, 11)  insert into DishIngredient values (3, 1)  insert into DishIngredient values (3, 3)  insert into DishIngredient values (3, 4)  insert into DishIngredient values (3, 5)  insert into DishIngredient values (3, 81)  insert into DishIngredient values (4, 5)  insert into DishIngredient values (5, 1)  insert into DishIngredient values (6, 1)  insert into DishIngredient values (6, 4)  insert into DishIngredient values (6, 16)  insert into DishIngredient values (6, 18)  insert into DishIngredient values (7, 7)  insert into DishIngredient values (7, 8)  insert into DishIngredient values (7, 9)  insert into DishIngredient values (7, 10)  insert into DishIngredient values (7, 11)  insert into DishIngredient values (7, 17)  insert into DishIngredient values (8, 4)  insert into DishIngredient values (8, 5)  insert into DishIngredient values (8, 11)  insert into DishIngredient values (8, 12)  insert into DishIngredient values (8, 13)  insert into DishIngredient values (8, 14)  insert into DishIngredient values (8, 16)  insert into DishIngredient values (9, 1)  insert into DishIngredient values (9, 4)  insert into DishIngredient values (9, 5)  insert into DishIngredient values (9, 14)  insert into DishIngredient values (9, 15)  insert into DishIngredient values (9, 16)  insert into DishIngredient values (10, 19)  insert into DishIngredient values (10, 17)  insert into DishIngredient values (10, 68)  insert into DishIngredient values (10, 11)  insert into DishIngredient values (11, 7)  insert into DishIngredient values (11, 20)  insert into DishIngredient values (11, 21)  insert into DishIngredient values (11, 22)  insert into DishIngredient values (11, 23)  insert into DishIngredient values (11, 24)  insert into DishIngredient values (11, 25)  insert into DishIngredient values (11, 26)  insert into DishIngredient values (12, 7)  insert into DishIngredient values (12, 11)  insert into DishIngredient values (12, 27)  insert into DishIngredient values (13, 7)  insert into DishIngredient values (13, 11)  insert into DishIngredient values (13, 27)  insert into DishIngredient values (14, 28)  insert into DishIngredient values (15, 29)  insert into DishIngredient values (15, 30)  insert into DishIngredient values (15, 31)  insert into DishIngredient values (15, 32)  insert into DishIngredient values (15, 33)  insert into DishIngredient values (15, 34)  insert into DishIngredient values (16, 4)  insert into DishIngredient values (16, 35)  insert into DishIngredient values (16, 36)  insert into DishIngredient values (17, 6)  insert into DishIngredient values (17, 12)  insert into DishIngredient values (17, 29)  insert into DishIngredient values (17, 35)  insert into DishIngredient values (18, 37)  insert into DishIngredient values (18, 5)  insert into DishIngredient values (18, 82)  insert into DishIngredient values (19, 4)  insert into DishIngredient values (19, 38)  insert into DishIngredient values (19, 35)  insert into DishIngredient values (19, 68)  insert into DishIngredient values (20, 4)  insert into DishIngredient values (20, 68)  insert into DishIngredient values (21, 1)  insert into DishIngredient values (21, 5)  insert into DishIngredient values (21, 39)  insert into DishIngredient values (21, 40)  insert into DishIngredient values (22, 39)  insert into DishIngredient values (22, 43)  insert into DishIngredient values (23, 41)  insert into DishIngredient values (23, 43)  insert into DishIngredient values (24, 42)  insert into DishIngredient values (24, 43)  insert into DishIngredient values (25, 7)  insert into DishIngredient values (25, 43)  insert into DishIngredient values (26, 42)  insert into DishIngredient values (26, 44)  insert into DishIngredient values (26, 45)  insert into DishIngredient values (26, 4)  insert into DishIngredient values (27, 39)  insert into DishIngredient values (27, 44)  insert into DishIngredient values (27, 45)  insert into DishIngredient values (27, 4)  insert into DishIngredient values (28, 44)  insert into DishIngredient values (28, 45)  insert into DishIngredient values (28, 4)  insert into DishIngredient values (29, 46)  insert into DishIngredient values (29, 5)  insert into DishIngredient values (30, 37)  insert into DishIngredient values (30, 33)  insert into DishIngredient values (31, 47)  insert into DishIngredient values (31, 50)  insert into DishIngredient values (31, 52)  insert into DishIngredient values (31, 6)  insert into DishIngredient values (31, 26)  insert into DishIngredient values (31, 48)  insert into DishIngredient values (31, 39)  insert into DishIngredient values (31, 68)  insert into DishIngredient values (32, 47)  insert into DishIngredient values (32, 50)  insert into DishIngredient values (32, 53)  insert into DishIngredient values (32, 5)  insert into DishIngredient values (32, 81)  insert into DishIngredient values (32, 6)  insert into DishIngredient values (32, 26)  insert into DishIngredient values (32, 48)  insert into DishIngredient values (32, 39)  insert into DishIngredient values (32, 68)  insert into DishIngredient values (33, 47)  insert into DishIngredient values (33, 50)  insert into DishIngredient values (33, 53)  insert into DishIngredient values (33, 5)  insert into DishIngredient values (33, 81)  insert into DishIngredient values (33, 52)  insert into DishIngredient values (33, 68)  insert into DishIngredient values (34, 47)  insert into DishIngredient values (34, 44)  insert into DishIngredient values (34, 4)  insert into DishIngredient values (34, 5)  insert into DishIngredient values (34, 81)  insert into DishIngredient values (34, 39)  insert into DishIngredient values (34, 68)  insert into DishIngredient values (35, 47)  insert into DishIngredient values (35, 44)  insert into DishIngredient values (35, 4)  insert into DishIngredient values (35, 5)  insert into DishIngredient values (35, 81)  insert into DishIngredient values (35, 82)  insert into DishIngredient values (35, 68)  insert into DishIngredient values (36, 54)  insert into DishIngredient values (36, 5)  insert into DishIngredient values (36, 37)  insert into DishIngredient values (36, 47)  insert into DishIngredient values (36, 81)  insert into DishIngredient values (37, 1)  insert into DishIngredient values (37, 5)  insert into DishIngredient values (37, 62)  insert into DishIngredient values (37, 55)  insert into DishIngredient values (37, 56)  insert into DishIngredient values (38, 1)  insert into DishIngredient values (38, 5)  insert into DishIngredient values (38, 62)  insert into DishIngredient values (38, 57)  insert into DishIngredient values (38, 58)  insert into DishIngredient values (38, 59)  insert into DishIngredient values (38, 60)  insert into DishIngredient values (38, 61)  insert into DishIngredient values (39, 1)  insert into DishIngredient values (39, 5)  insert into DishIngredient values (39, 62)  insert into DishIngredient values (39, 63)  insert into DishIngredient values (39, 64)  insert into DishIngredient values (40, 65)  insert into DishIngredient values (40, 66)  insert into DishIngredient values (40, 67)  insert into DishIngredient values (40, 68)  insert into DishIngredient values (40, 31)  insert into DishIngredient values (41, 1)  insert into DishIngredient values (41, 5)  insert into DishIngredient values (41, 31)  insert into DishIngredient values (41, 69)  insert into DishIngredient values (41, 70)  insert into DishIngredient values (42, 1)  insert into DishIngredient values (42, 5)  insert into DishIngredient values (42, 71)  insert into DishIngredient values (42, 72)  insert into DishIngredient values (42, 73)  insert into DishIngredient values (42, 74)  insert into DishIngredient values (43, 5)  insert into DishIngredient values (43, 7)  insert into DishIngredient values (43, 26)  insert into DishIngredient values (43, 75)  insert into DishIngredient values (43, 76)  insert into DishIngredient values (44, 1)  insert into DishIngredient values (44, 5)  insert into DishIngredient values (44, 41)  insert into DishIngredient values (44, 51)  insert into DishIngredient values (44, 77)  insert into DishIngredient values (45, 1)  insert into DishIngredient values (45, 5)  insert into DishIngredient values (45, 39)  insert into DishIngredient values (45, 51)  insert into DishIngredient values (45, 77)  insert into DishIngredient values (46, 1)  insert into DishIngredient values (46, 5)  insert into DishIngredient values (46, 13)  insert into DishIngredient values (46, 51)  insert into DishIngredient values (46, 77)  insert into DishIngredient values (46, 78)  insert into DishIngredient values (47, 83)  insert into DishIngredient values (47, 51)  insert into DishIngredient values (47, 77)  insert into DishIngredient values (47, 82)  insert into DishIngredient values (47, 35)  insert into DishIngredient values (48, 84)  insert into DishIngredient values (48, 79)  insert into DishIngredient values (48, 80)  insert into DishIngredient values (48, 58)  insert into DishIngredient values (48, 75)  insert into DishIngredient values (48, 22) |

# Display tables

## Passenger

|  |
| --- |
| Select \* from Passenger; |



## ContactNo

|  |
| --- |
| Select \* from ContactNo; |

Table

Description automatically generated with medium confidence

## EventType

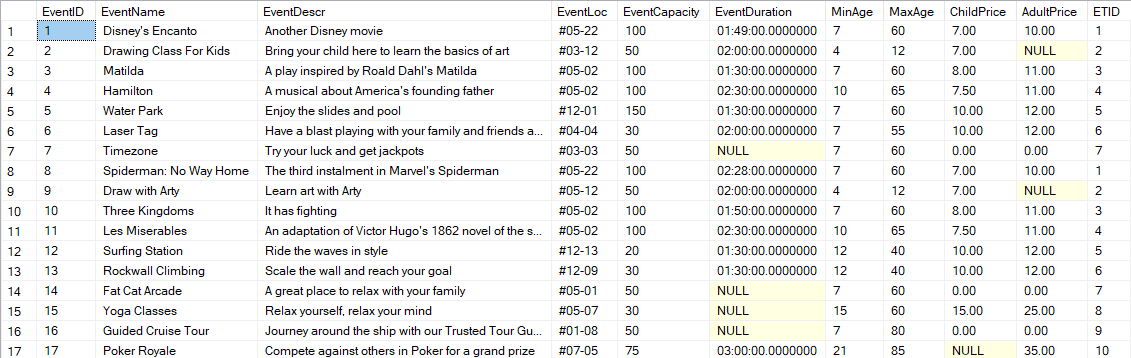
|  |
| --- |
| Select \* from EventType; |

Table

Description automatically generated with medium confidence

## Event

|  |
| --- |
| Select \* from Event; |



## EventSession

|  |
| --- |
| Select \* from EventSession; |

Table

Description automatically generated

## Booking

|  |
| --- |
| Select \* from Booking; |

Table

Description automatically generated with medium confidence

## Eatery

|  |
| --- |
| Select \* from Eatery; |

Table

Description automatically generated

## Reservation

|  |
| --- |
| Select \* from Reservation; |

Table

Description automatically generated with medium confidence

## Cuisine

|  |
| --- |
| Select \* from Cuisine; |

Table

Description automatically generated

## Dish

|  |
| --- |
| Select \* from Dish; |

A screenshot of a computer

Description automatically generated with low confidence

A picture containing text, document, screenshot

Description automatically generated

## CSDish

|  |
| --- |
| Select \* from CSDish; |

Table

Description automatically generated

Table

Description automatically generated

## CSOrder

|  |
| --- |
| Select \* from CSOrder; |

Table

Description automatically generated

Table

Description automatically generated

## FoodCategory

|  |
| --- |
| Select \* from FoodCategory; |

Graphical user interface, text, application, table

Description automatically generated

## DishFoodCategory

|  |
| --- |
| Select \* from DishFoodCategory; |

Table

Description automatically generated with low confidence

A picture containing table

Description automatically generated

A picture containing table

Description automatically generated

## Ingredient

|  |
| --- |
| Select \* from Ingredient; |

Table

Description automatically generated

Table

Description automatically generated with medium confidence

Table

Description automatically generated with medium confidence

Table

Description automatically generated with low confidence

## DishIngredient

|  |
| --- |
| Select \* from DishIngredient; |

Chart

Description automatically generated with medium confidence

A picture containing calendar

Description automatically generated

A picture containing calendar

Description automatically generated

A picture containing table

Description automatically generated

A picture containing table

Description automatically generated

A picture containing table

Description automatically generated

A picture containing table

Description automatically generated

Table

Description automatically generated

# Queries of each member

## Kelven

1. List the number of bookings made for each event

Purpose: To find out if which event is more popular

|  |
| --- |
| select count(b.BookingID) as 'No. of bookings', e.EventName  from Booking b inner join Event e on b.EventID = e.EventID  group by e.EventName |

Table

Description automatically generated

1. List the number of dishes ordered via cabin service from each cuisine

Purpose: To find out which cuisine passengers like to eat

|  |
| --- |
| select count(cs.DishID) as 'No. of dishes', c.CuisineName  from CSOrder cs inner join Dish d on cs.DishID = d.DishID  inner join Cuisine c on d.CuisineID = c.CuisineID  group by c.CuisineName |

Table

Description automatically generated with medium confidence

1. List the ID and name of eateries that have the least sales

Purpose: To find which eatery is doing poorly in sales

|  |
| --- |
| select e.EatyID, e.EatyName, sum(OrderPrice) as 'Sales'  from Eatery e, CSOrder cs, Dish d  where cs.DishID in (select d.DishID where d.EatyID = e.EatyID)  group by e.EatyID, e.EatyName  order by 'Sales' asc |

Table

Description automatically generated

## Yue Ming

1. List the number of dishes in each food category for each restaurant

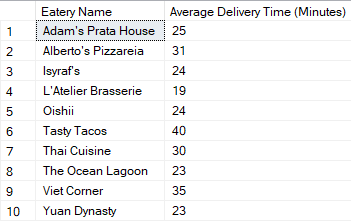
Purpose: To find out what restaurants offers what type of food

|  |
| --- |
| select e.EatyName as ‘Eatery Name’, fc.FcName, count(dfc.DishID) as 'No. of food type'  from Eatery e inner join Dish d on e.EatyID = d.EatyID  inner join DishFoodCategory dfc on dfc.DishID = d.DishID  inner join FoodCategory fc on fc.FcID = dfc.FcID  group by fc.FcName, e.EatyName |



1. List the Eatery Name and Average Delivery Time of each restaurant.  
   Purpose: To show the average delivery rate of each restaurant

|  |
| --- |
| select EatyName as 'Eatery Name', (avg(DATEDIFF(minute, OrderDateTime, DelDateTime))) as 'Average Delivery Time (Minutes)' from CSOrder  inner join Dish on Dish.DishID = CSOrder.DishID  inner join Eatery on Eatery.EatyID = Dish.EatyID  group by EatyName |



1. List the EatyName and Seats Occupied in each restaurant.   
   Purpose: To check the number seats that were reserved in each restaurant during the journey of the cruise

|  |
| --- |
| select e.EatyName as 'Eatery Name', sum(r.NoOfPax) as 'Seats occupied'  from Eatery e inner join Reservation r on e.EatyID = r.EatyID  group by e.EatyName, e.EatyCapacity  order by EatyName asc |

