Entities:

Passenger: This entity represents passengers who book flights. Attributes include:

pnr_no (primary key)

name

address

nationality

gender

ph_no

passport_no

Flight: This entity represents flights offered by the airline. Attributes include:

f_code (primary key)

f_name

src (source airport)

dst (destination airport)

Reservation: This entity represents a passenger's reservation for a specific flight. Attributes include:

pnr_no (foreign key referencing Passenger)

ticket_id (primary key)

f_code (foreign key referencing Flight)

jny_date (journey date)

jny_time (journey time)

src (inherited from Flight)

dst (inherited from Flight)

Cancellation: This entity represents cancellations of reservations. Attributes include:

pnr_no (foreign key referencing Passenger)

cancellation_no (primary key)

cancellation_date

fli_code (foreign key referencing Flight) Note: This might be a typo and should be f_code for consistency.

Payment: This entity represents payments made for reservations. Attributes include:

pnr_no (foreign key referencing Passenger)

ph_no

cheque_no

card_no

paid_amt

pay_date

Sector: This entity represents different classes (seating categories) available on a flight. Attributes include:

flight_code (foreign key referencing Flight)

capacity

class_code (primary key)

class_name

Relationships:

One Passenger can have Many Reservations: A passenger can have multiple reservations (one-to-many). This is represented by the pnr_no foreign key in the Reservation table referencing the Passenger table.

One Flight can have Many Reservations: A flight can have many reservations (one-to-many). This is represented by the f_code foreign key in the Reservation table referencing the Flight table.

One Reservation belongs to One Passenger: A reservation is for a single passenger (many-to-one). This is the inverse of the one-to-many relationship between Passenger and Reservation.

One Reservation is for One Flight: A reservation is for a specific flight (many-to-one). This is the inverse of the one-to-many relationship between Flight and Reservation.

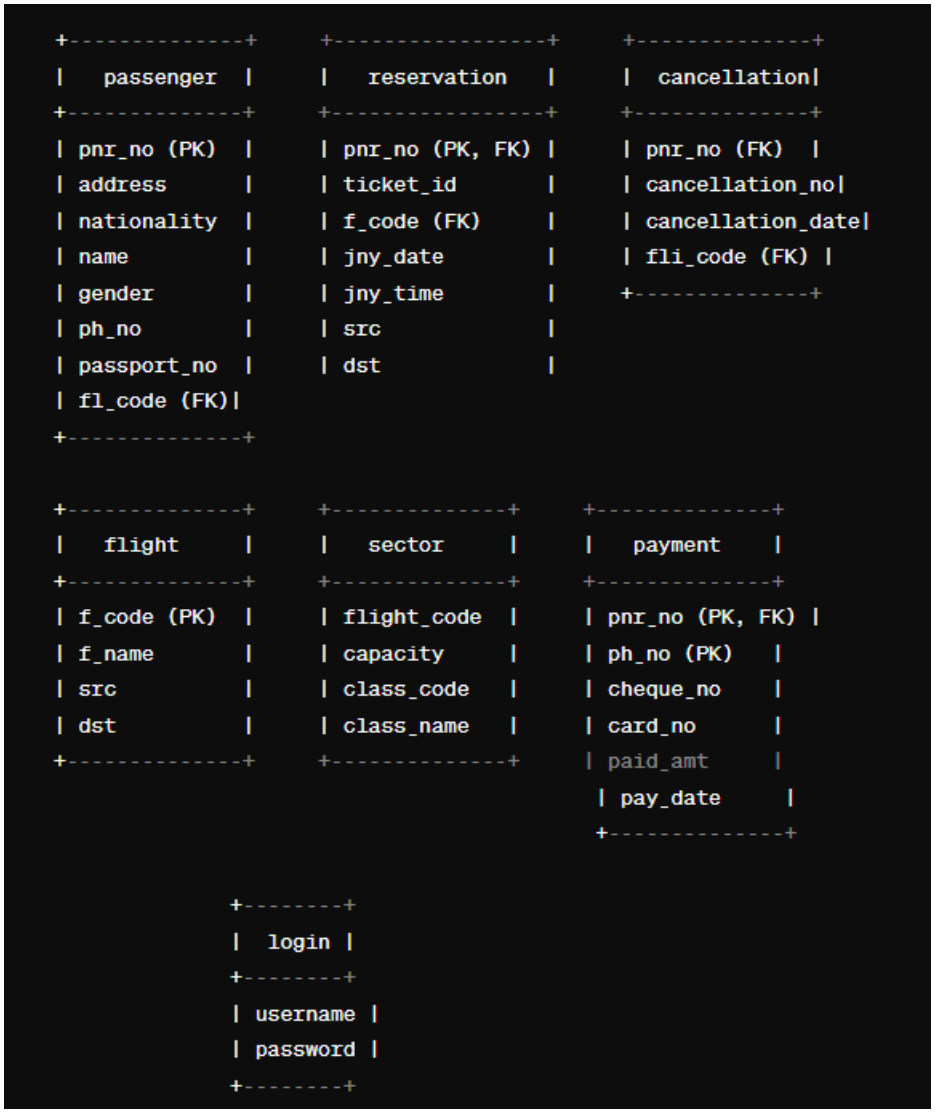
One Passenger can have One Cancellation (or None): A passenger might have a cancellation for their reservation (one-to-one or one-to-zero). This is represented by the pnr_no foreign key in the Cancellation table referencing the Passenger table.

One Flight can have Many Cancellations: A flight can have many cancellations associated with reservations (one-to-many). This is represented by the f_code (assuming it's a typo and should be f_code) foreign key in the Cancellation table referencing the Flight table. Note: This might need correction based on the actual database design.

One Passenger can have One Payment (or None): A passenger might have a payment made for their reservation (one-to-one or one-to-zero). This is represented by the pnr_no foreign key in the Payment table referencing the Passenger table.

One Flight has Many Sectors: A flight can have multiple sectors representing different classes (one-to-many). This is represented by the flight_code foreign key in the Sector table referencing the Flight table.

I hope this ER diagram accurately reflects the relationships between the entities in your airline reservation system.


passenger
pnr_no (PK)
address
nationality
name
gender
ph_no
passport_no
fl_code (FK)
reservation
pnr_no (PK, FK)
ticket_id
f_code (FK)
jny_date
jny_time
src
dst
cancellation
pnr_no (FK)
cancellation_no
cancellation_date
fli_code (FK)
flight
f_code (PK)
f_name
src
dst
sector
flight_code
capacity
class_code
class_name
payment
pnr_no (PK, FK)
ph_no (PK)
cheque_no
card_no
paid_amt
pay_date
login
username
password

CODE-

```
create database project4;


use project4;

----------------------------------------------


Table structure for table `cancellation`

--


CREATE TABLE `cancellation` (
  `pnr_no` varchar(30) NOT NULL,
  `cancellation_no` varchar(30) NOT NULL,
  `cancellation_date` varchar(30) NOT NULL,
  `fli_code` varchar(30) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;



INSERT INTO `cancellation` (`pnr_no`, `cancellation_no`, `cancellation_date`, `fli_code`) VALUES
('2', '1005', '22-05-05', 'f1005');


select * from cancellation;

-- ----------------------------------------------------------


--

-- Table structure for table `flight`
```

```
--


CREATE TABLE `flight` (
  `f_code` varchar(30) NOT NULL,
  `f_name` varchar(30) NOT NULL,
  `src` varchar(30) NOT NULL,
  `dst` varchar(30) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;


--
-- Dumping data for table `flight`
--
INSERT INTO `flight` (`f_code`, `f_name`, `src`, `dst`) VALUES
('f1003', 'flight 1001', 'Australia', 'United States');
select * from flight;
-- ------------------------------------------------------
delete f_code from flight where f_code=f1005;
update flight set
--
-- Table structure for table `login`
--


CREATE TABLE `login` (
  `username` varchar(30) NOT NULL,
  `password` varchar(30) NOT NULL
```

```
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;


--

-- Dumping data for table `login`

--


INSERT INTO `login` (`username`, `password`) VALUES

('admin', 'admin');


-- --------------------------------------------------------


--

-- Table structure for table `passenger`

--

select * from passenger;

CREATE TABLE `passenger` (

  `pnr_no` varchar(30) NOT NULL,

  `address` text NOT NULL,

  `nationality` varchar(30) NOT NULL,

  `name` varchar(30) NOT NULL,

  `gender` varchar(30) NOT NULL,

  `ph_no` varchar(30) NOT NULL,

  `passport_no` varchar(30) NOT NULL,

  `fl_code` varchar(30) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;
```

```
--

-- Dumping data for table `passenger`

--


INSERT INTO `passenger` (`pnr_no`, `address`, `nationality`, `name`, `gender`, `ph_no`, `passport_no`, `fl_code`) VALUES

('2000', 'Negros Occidental, Philippines', 'Filipino', '', 'male', '09272777334', 'SAR081119', 'f1005');


-- --------------------------------------------------------


--

-- Table structure for table `payment`

--


CREATE TABLE `payment` (

  `pnr_no` varchar(30) NOT NULL,

  `ph_no` varchar(30) NOT NULL,

  `cheque_no` varchar(30) NOT NULL,

  `card_no` varchar(30) NOT NULL,

  `paid_amt` varchar(30) NOT NULL,

  `pay_date` varchar(30) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;


--
```

```
-- Dumping data for table `payment`

--


INSERT INTO `payment` (`pnr_no`, `ph_no`, `cheque_no`, `card_no`, `paid_amt`, `pay_date`) VALUES

('9', '7654606488', '1002', '162-401', '1070', '2023-1-22');

select * from payment;

-- --------------------------------------------------------


--

-- Table structure for table `reservation`

--


CREATE TABLE `reservation` (

  `pnr_no` varchar(30) NOT NULL,

  `ticket_id` varchar(30) NOT NULL,

  `f_code` varchar(30) NOT NULL,

  `jny_date` varchar(30) NOT NULL,

  `jny_time` varchar(30) NOT NULL,

  `src` varchar(30) NOT NULL,

  `dst` varchar(30) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;


--

-- Dumping data for table `reservation`

--
```

```
select * from reservation;

INSERT INTO `reservation` (`pnr_no`, `ticket_id`, `f_code`, `jny_date`, `jny_time`, `src`, `dst`) VALUES

('12', '101', 'f1004', '2022-05-05', '3:00 PM', 'Australia', 'Russia');

select * from reservation;

-- --------------------------------------------------------



--

-- Table structure for table `sector`

--



CREATE TABLE `sector` (

  `flight_code` varchar(30) NOT NULL,

  `capacity` varchar(30) NOT NULL,

  `class_code` varchar(30) NOT NULL,

  `class_name` varchar(30) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;



--

-- Dumping data for table `sector`

--



INSERT INTO `sector` (`flight_code`, `capacity`, `class_code`, `class_name`) VALUES

('f1004', '1000', 'A', 'FIRST CLASS'),

('f1004', '700', 'B', 'BUSINESS CLASS'),

('f1004', '500', 'C', 'ECONOMY');
```

```
select * from sector;

COMMIT;


/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;

/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;

/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;

select * from pasenger;

select * from project4;

select * from sector;
```