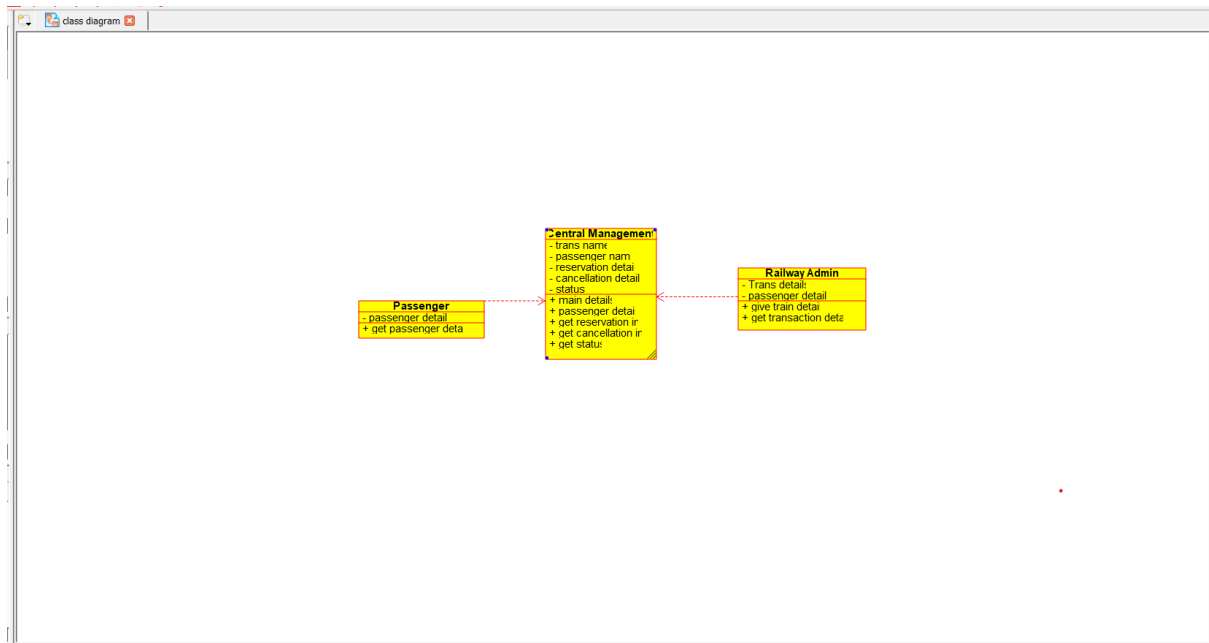
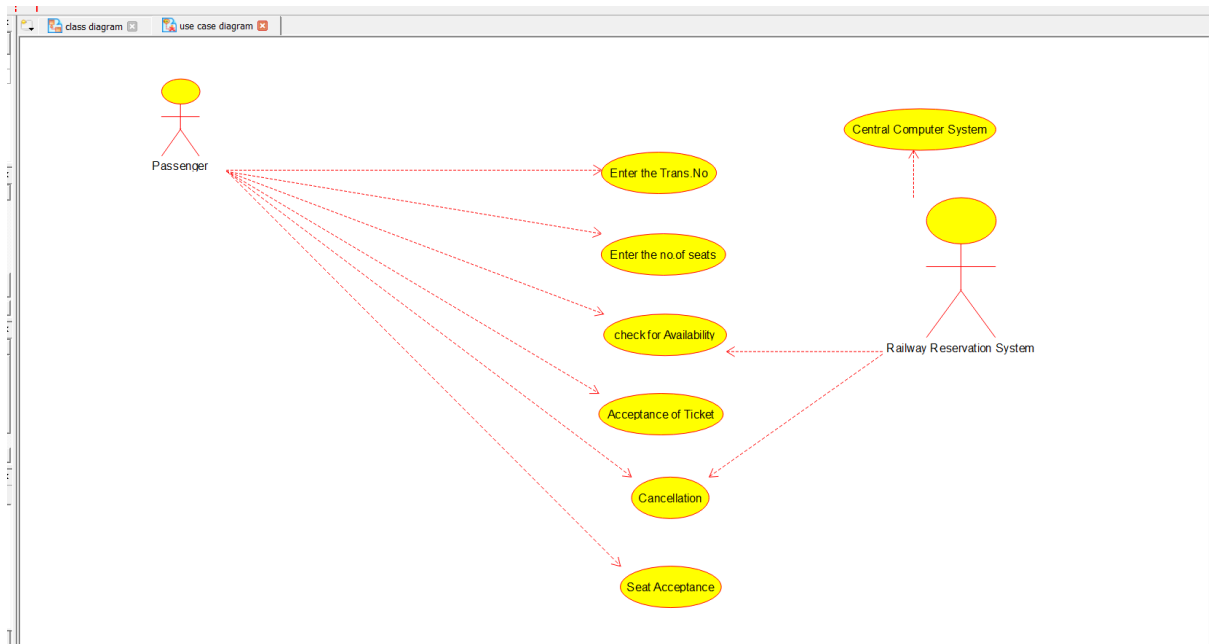


F TICKETING



```
#ifndef PASSENGER_H
```

```
#define PASSENGER_H
```

```
#include <string>
```

```
/**
```

```
 * class Passenger
```

```
 *
```

```
 */
```

```
class Passenger
```

```
{
```

```
public:
```

```
    // Constructors/Destructors
```

```
    //
```

```
/**
```

```
 * Empty Constructor
```

```
 */
```

```
    Passenger ();
```

```
/**
```

```
 * Empty Destructor
```

```
 */
```

```
    virtual ~Passenger ();
```

```
    // Static Public attributes
```

```
    //
```

```
// Public attributes
```

```
//
```

```
// Public attribute accessor methods
```

```
//
```

```
// Public attribute accessor methods
```

```
//
```

```
/**
```

```
*/
```

```
void get_passenger_details ()
```

```
{
```

```
}
```

```
protected:
```

```
// Static Protected attributes
```

```
//
```

```
// Protected attributes
```

```
//
```

```
public:
```

```
// Protected attribute accessor methods
```

```
//
```

```
protected:
```

```
public:
```

```
// Protected attribute accessor methods
```

```
//
```

```
protected:
```

```
private:
```

```
// Static Private attributes
```

```
//
```

```
// Private attributes
```

```
//
```

```
void passenger_details;
```

```
public:
```

```
// Private attribute accessor methods
```

```
//
```

```
private:
```

```
public:
```

```

// Private attribute accessor methods
//

/**
 * Set the value of passenger_details
 * @param new_var the new value of passenger_details
 */
void setPassenger_details (void new_var) {
    passenger_details = new_var;
}

/**
 * Get the value of passenger_details
 * @return the value of passenger_details
 */
void getPassenger_details () {
    return passenger_details;
}
private:

void initAttributes () ;

};

#endif // PASSENGER_H

```

```
#ifndef CENTRAL_MANAGEMENT_H
```

```
#define CENTRAL_MANAGEMENT_H
```

```
#include <string>
```

```
/**
```

```
 * class Central_Management
```

```
 *
```

```
 */
```

```
class Central_Management
```

```
{
```

```
public:
```

```
    // Constructors/Destructors
```

```
    //
```

```
/**
```

```
 * Empty Constructor
```

```
 */
```

```
Central_Management ();
```

```
/**
```

```
 * Empty Destructor
```

```
 */
```

```
virtual ~Central_Management ();
```

```
    // Static Public attributes
```

```
    //
```

```
// Public attributes
```

```
//
```

```
// Public attribute accessor methods
```

```
//
```

```
// Public attribute accessor methods
```

```
//
```

```
/**
```

```
*/
```

```
void main_details ()
```

```
{
```

```
}
```

```
/**
```

```
*/
```

```
void passenger_details ()
```

```
{
```

```
}
```

```
/**
```

```
*/
```

```
void get_reservation_info ()
```

```
{
```

```
}
```

```
/**
```

```
*/
```

```
void get_cancellation_info ()
```

```
{
```

```
}
```

```
/**
```

```
*/
```

```
void get_status ()
```

```
{
```

```
}
```

```
protected:
```

```
// Static Protected attributes
```

```
//
```

```
// Protected attributes
```

```
//
```

```
public:
```

```
// Protected attribute accessor methods
```

```
//
```

```
protected:
```


public:

// Protected attribute accessor methods

//

protected:

private:

// Static Private attributes

//

// Private attributes

//

void trans_name;

void passenger_name;

void reservation_details;

void cancellation_details_;

void status;

public:

// Private attribute accessor methods

//

private:

public:

// Private attribute accessor methods

//

/**

* Set the value of trans_name

* @param new_var the new value of trans_name

*/

void setTrans_name (void new_var) {

trans_name = new_var;

}

/**

* Get the value of trans_name

* @return the value of trans_name

*/

void getTrans_name () {

return trans_name;

}

/**

* Set the value of passenger_name

* @param new_var the new value of passenger_name

*/

void setPassenger_name (void new_var) {

passenger_name = new_var;

}

```
/**
 * Get the value of passenger_name
 * @return the value of passenger_name
 */
void getPassenger_name () {
    return passenger_name;
}
```

```
/**
 * Set the value of reservation_details
 * @param new_var the new value of reservation_details
 */
void setReservation_details (void new_var) {
    reservation_details = new_var;
}
```

```
/**
 * Get the value of reservation_details
 * @return the value of reservation_details
 */
void getReservation_details () {
    return reservation_details;
}
```

```
/**
 * Set the value of cancellation_details_
 * @param new_var the new value of cancellation_details_
 */
void setCancellation_details_ (void new_var) {
    cancellation_details_ = new_var;
}
```

```
/**
 * Get the value of cancellation_details_
 * @return the value of cancellation_details_
 */
void getCancellation_details_ () {
    return cancellation_details_;
}
```

```
/**
 * Set the value of status
 * @param new_var the new value of status
 */
void setStatus (void new_var) {
    status = new_var;
}
```

```
/**
 * Get the value of status
 * @return the value of status
 */
void getStatus () {
    return status;
}
```

private:

```
void initAttributes () ;
```

```
};
```

```
#endif // CENTRAL_MANAGEMENT_H
```

```
#ifndef RAILWAY_ADMIN_H
```

```
#define RAILWAY_ADMIN_H
```

```
#include <string>
```

```
/**
```

```
 * class Railway_Admin
```

```
 *
```

```
 */
```

```
class Railway_Admin
```

```
{
```

```
public:
```

```
    // Constructors/Destructors
```

```
    //
```

```
/**
```

```
 * Empty Constructor
```

```
 */
```

```
Railway_Admin ();
```

```
/**
```

```
 * Empty Destructor
```

```
 */
```

```
virtual ~Railway_Admin ();
```

```
// Static Public attributes
```

```
//
```

```
// Public attributes
```

```
//
```

```
// Public attribute accessor methods
```

```
//
```

```
// Public attribute accessor methods
```

```
//
```

```
/**
```

```
*/
```

```
void give_train_details ()
```

```
{
```

```
}
```

```
/**
```

```
*/
```

```
void get_transaction_details ()
```

```
{
```

```
}
```

```
protected:
```

```
// Static Protected attributes
```

```
//
```

```
// Protected attributes
```

```
//
```

```
public:
```

```
// Protected attribute accessor methods
```

```
//
```

```
protected:
```

```
public:
```

```
// Protected attribute accessor methods
```

```
//
```

```
protected:
```

```
private:
```

```
// Static Private attributes
```

```
//
```

```
// Private attributes
```

```
//
```

```
void Trans_details;

void passenger_details;

public:


// Private attribute accessor methods
//

private:


public:


// Private attribute accessor methods
//


/**
 * Set the value of Trans_details
 * @param new_var the new value of Trans_details
 */
void setTrans_details (void new_var) {
    Trans_details = new_var;
}


/**
 * Get the value of Trans_details
 * @return the value of Trans_details
 */
void getTrans_details () {
    return Trans_details;
}
```



```
}
```

```
/**
```

```
 * Set the value of passenger_details
```

```
 * @param new_var the new value of passenger_details
```

```
 */
```

```
void setPassenger_details (void new_var) {
```

```
    passenger_details = new_var;
```

```
}
```

```
/**
```

```
 * Get the value of passenger_details
```

```
 * @return the value of passenger_details
```

```
 */
```

```
void getPassenger_details () {
```

```
    return passenger_details;
```

```
}
```

```
private:
```

```
void initAttributes () ;
```

```
};
```

```
#endif // RAILWAY_ADMIN_H
```