a) Write a constructor for the class Movie, which takes a String representing the title of the movie, a String representing the studio, and a String representing the rating as its arguments, and sets the respective class properties to these values.

class movie {

constructor(title,studio,rating) {

this.movie=title;

this.studio=studio;

this.rating=rating;

}

}

movie = new movie("FAST AND FURIOUS","UNIVERSAL STUDIOS","PG13");

console.log(movie)

b) The constructor for the class Movie will set the class property rating to "PG" as default when no rating is provided.

class movie {

constructor(title,studio,rating="PG") {

this.movie=title;

this.studio=studio;

this.rating=rating;

}

}

movie = new movie("FAST AND FURIOUS "," PG13");

console.log(movie.rating);

c) Write a method getPG, which takes an array of base type Movie as its argument, and returns a new array of only those movies in the input array with a rating of "PG". You may assume the input array is full of Movie instances. The returned array need not be full.

class movie {

constructor(title,studio,rating) {

this.movie=title;

this.studio=studio;

this.rating=rating;

}

}

movie1 = new movie("FAST AND FURIOUS","UNIVERSAL STUDIOS","PG13");

movie2 =new movie ("AQUAMAN","DC","PG13");

movie3 = new movie ("SPIDERMAN","MARVEL","PG11")

console.log(movie2)

d) Write a piece of code that creates an instance of the class Movie with the title “Casino Royale”, the studio “Eon Productions”, and the rating “PG­13”

class movie {

constructor(title,studio,rating) {

this.movie=title;

this.studio=studio;

this.rating=rating;

}

}

movie = new movie("Casino Royale","Eon Productions","PG13");

console.log(movie)

2. Convert the UML diagram to Typescript class. - use number for double

class circle {

constructor(double,string) {

this.radius=double;

this.color=string;

}

get circle() {

return circle;

}

get circle1(radius=1.0){

return ;

}

get circle2 (radius=1.0,color="red"){

return ;

}

get getRadius(){

return this.radius;

}

get setRadius(radius=1.0){

return void circle1();

}

get getColor(){

return this.color;

}

get setColor(color=string){

return void circle2();

}

get toString(){

return `circle[radius=${double},color=${string}]`;

}

get getArea(){

return 3.14 \* double \* double;

}

get getCircumference(){

return 2\* 3.14 \* double;

}

}

const t = new uberPrice(1.0,"red");

console.log(t.getArea)

3. Write a “person” class to hold all the details

class Person{

constructor(name,phonenumber,Email,address){

this.name=name;

this.contact=phonenumber;

this.mail=Email;

this.address=address;

}

}

person=new Person("muthu",9566420177,"mkumar@gmail.com","pudukkottai");

console.log(person)

4. write a class to calculate uber price.

class uberPrice {

constructor(quantity,price) {

this.no = quantity;

this.price = price;

}

get total () {

return this.no \* this.price;

}

}

const t = new uberPrice(2,100);

console.log(t.total);