Advanced Programming Assignment-2 (RUBRIC)		
Criteria	**Description**	**Marks**
Correct Implementation	We can enter the program in two ways - as the admin or as a visitor. The code should give the correct output for the mentioned functionalities. Binary marking is to be followed. No marks are to be given for partially correct outputs. The output format can vary a little, but it should be complete. Note that small assumptions regarding discounts and attractions by students are permissible as long as they can explain it	2
	The submission should have a ReadMe, POM.xml and relevant code files and be compiled using Maven. Good naming conventions should be followed for classes, data members, other variables and methods	1
Functionality	The code should have a sign-in feature for customers and sign-in for admins using the correct password, note that no sign-in should happen with an incorrect password	1
	The admin functionalities work as expected, including managing attractions and animals, setting discounts and special deals, and1 viewing visitor stats and feedback.	2
	The visitor functionalities work as expected, including browsing different attractions and animals at the zoo without visiting them, visiting the animals, buying membership or tickets, visiting attractions, and providing feedback.	2
Inheritance	The code should exhibit inheritance. There can be many places where this can be done, like, For example, the PremiumVisitor class inherits from the Visitor class. Another example can be the class Snake(can be another animal too) which inherits from Reptiles which in turn inherits from the abstract class Animal If the student has any form of IS-A relationship which they can justify, give marks.	3
Encapsulation / correct use of modifiers	Data members must have private access unless it's absolutely necessary to grant them public/default access.	2
	Appropriate Getters and Setters should be used.	1

Advanced Programming Assignment-2 (RUBRIC)		
Criteria	**Description**	**Marks**
Correct Implementation	We can enter the program in two ways - as the admin or as a visitor. The code should give the correct output for the mentioned functionalities. Binary marking is to be followed. No marks are to be given for partially correct outputs. The output format can vary a little, but it should be complete. Note that small assumptions regarding discounts and attractions by students are permissible as long as they can explain it	2
	The submission should have a ReadMe, POM.xml and relevant code files and be compiled using Maven. Good naming conventions should be followed for classes, data members, other variables and methods	1
Functionality	The code should have a sign-in feature for customers and sign-in for admins using the correct password, note that no sign-in should happen with an incorrect password	1
	The admin functionalities work as expected, including managing attractions and animals, setting discounts and special deals, and1 viewing visitor stats and feedback.	2
	The visitor functionalities work as expected, including browsing different attractions and animals at the zoo without visiting them, visiting the animals, buying membership or tickets, visiting attractions, and providing feedback.	2
Inheritance	The code should exhibit inheritance. There can be many places where this can be done, like, For example, the PremiumVisitor class inherits from the Visitor class. Another example can be the class Snake(can be another animal too) which inherits from Reptiles which in turn inherits from the abstract class Animal If the student has any form of IS-A relationship which they can justify, give marks.	3
Interface / Abstract class	The code should have at least one interface/ abstract class, for example, there can be many possibilities here such as an interface to buy tickets or visit attractions each visitor type implements it differently. Aside this accept any interface as long as it serves as a contract for the classes it is implemented by and students can	3

Advanced Programming Assignment-2 (RUBRIC)		
Criteria	**Description**	**Marks**
Correct Implementation	We can enter the program in two ways - as the admin or as a visitor. The code should give the correct output for the mentioned functionalities. Binary marking is to be followed. No marks are to be given for partially correct outputs. The output format can vary a little, but it should be complete. Note that small assumptions regarding discounts and attractions by students are permissible as long as they can explain it	2
	The submission should have a ReadMe, POM.xml and relevant code files and be compiled using Maven. Good naming conventions should be followed for classes, data members, other variables and methods	1
Functionality	The code should have a sign-in feature for customers and sign-in for admins using the correct password, note that no sign-in should happen with an incorrect password	1
	The admin functionalities work as expected, including managing attractions and animals, setting discounts and special deals, and1 viewing visitor stats and feedback.	2
	The visitor functionalities work as expected, including browsing different attractions and animals at the zoo without visiting them, visiting the animals, buying membership or tickets, visiting attractions, and providing feedback.	2
Inheritance	The code should exhibit inheritance. There can be many places where this can be done, like, For example, the PremiumVisitor class inherits from the Visitor class. Another example can be the class Snake(can be another animal too) which inherits from Reptiles which in turn inherits from the abstract class Animal If the student has any form of IS-A relationship which they can justify, give marks.	3
	provide a justification for the same.	
Polymorphism	The code should exhibit polymorphism in the case of visitor membership types - an object of type Basic/Premium can be referred to using a variable of type Membership, but on calling a function, the implementation specific to the actual class	3

Advanced Programming Assignment-2 (RUBRIC)		
Criteria	**Description**	**Marks**
Correct Implementation	We can enter the program in two ways - as the admin or as a visitor. The code should give the correct output for the mentioned functionalities. Binary marking is to be followed. No marks are to be given for partially correct outputs. The output format can vary a little, but it should be complete. Note that small assumptions regarding discounts and attractions by students are permissible as long as they can explain it	2
	The submission should have a ReadMe, POM.xml and relevant code files and be compiled using Maven. Good naming conventions should be followed for classes, data members, other variables and methods	1
Functionality	The code should have a sign-in feature for customers and sign-in for admins using the correct password, note that no sign-in should happen with an incorrect password	1
	The admin functionalities work as expected, including managing attractions and animals, setting discounts and special deals, and1 viewing visitor stats and feedback.	2
	The visitor functionalities work as expected, including browsing different attractions and animals at the zoo without visiting them, visiting the animals, buying membership or tickets, visiting attractions, and providing feedback.	2
Inheritance	The code should exhibit inheritance. There can be many places where this can be done, like, For example, the PremiumVisitor class inherits from the Visitor class. Another example can be the class Snake(can be another animal too) which inherits from Reptiles which in turn inherits from the abstract class Animal If the student has any form of IS-A relationship which they can justify, give marks.	3
	(Basic/Premium) should run Another example can be for visiting the attractions or the animals - the visitAttraction() method can be used to visit any type of attraction, regardless of its specific class, similarly for the animals. If students have shown polymorphism in any other manner, give marks as long as they justify it and show	

Advanced Programming Assignment-2 (RUBRIC)		
Criteria	**Description**	**Marks**
Correct Implementation	We can enter the program in two ways - as the admin or as a visitor. The code should give the correct output for the mentioned functionalities. Binary marking is to be followed. No marks are to be given for partially correct outputs. The output format can vary a little, but it should be complete. Note that small assumptions regarding discounts and attractions by students are permissible as long as they can explain it	2
	The submission should have a ReadMe, POM.xml and relevant code files and be compiled using Maven. Good naming conventions should be followed for classes, data members, other variables and methods	1
Functionality	The code should have a sign-in feature for customers and sign-in for admins using the correct password, note that no sign-in should happen with an incorrect password	1
	The admin functionalities work as expected, including managing attractions and animals, setting discounts and special deals, and1 viewing visitor stats and feedback.	2
	The visitor functionalities work as expected, including browsing different attractions and animals at the zoo without visiting them, visiting the animals, buying membership or tickets, visiting attractions, and providing feedback.	2
Inheritance	The code should exhibit inheritance. There can be many places where this can be done, like, For example, the PremiumVisitor class inherits from the Visitor class. Another example can be the class Snake(can be another animal too) which inherits from Reptiles which in turn inherits from the abstract class Animal If the student has any form of IS-A relationship which they can justify, give marks.	3
	a working example. Follow binary marking.	
	Total	20