PRN No.

PAPER CODE

May 2024 (ENDSEM) EXAM

F.Y.B. TECH. (SEMESTER - II)

COURSE NAME: Programming And Problem Solving-II COURSE CODE: SE12233 Branch: CESE

(PATTERN 2023)

Time: [1Hr. 30 Min]

[Max. Marks: 40]

- () Instructions to candidates:
- 1) Figures to the right indicate full marks.
- 2) Use of scientific calculator is allowed
- 3) Use suitable data wherever required
- 4) All questions are compulsory. Solve any one sub question from each Question 1 and 2 and any three sub questions each from Questions 3 and 4.

	O the Description	Max.	СО	BT Level
Q. No.	Question Description	Marks	mapped	
Q.1	a) Describe how Java language is platform independent.	[5]	CO1	Understand
Q.1	a) Describe new extra sample 1			
	b) Describe the decision-making statements used in java	[5]	CO1	Understand
	with example.	rest	-	A1
Q2 (Apply the following operations on this	[5]	CO2	Apply
	'Good Morning Everyone' strings in a Java program:			
	 Calculate the length of the string. 			
	 Convert the string to uppercase. 			
	Convert the string to lowercase.			
	Substitute every occurrence of 'o' with 'i'.			
	Identify the characters located at the 5th and 13th			
	positions.			
	I Develop a Java	[5]	CO2	Apply
	b) Utilize a switch-case statement to Develop a Java	1 ' '		
	program that analyzes a given character to determine if			
	it falls into the category of vowels or consonants. a) Apply the concept of "this" keyword to distinguish	[5]	CO3	Apply
Q.3	between "local variable" and "instance variable" with			
	l control of the cont			
	example.			1
		[5]	CO3	Apply
	b) Apply method overloading within a Java program to)		
	create a class featuring two 'mul' methods capable of	f		
	performing multiplication operations on both integer and	i		
	floating-point number parameters.			

bike_name and regno while Employee class manages details such as emp_name and emp_id, and maintains a reference to a Bike object. Instantiate objects to illustrate the relationship, and display the employee's name and ID, along with the bike's name and registration number. d) Apply the 'super' keyword in Java to access parent class members within a child class. Parent initializes 'a' as 20, child class extends it, setting 'a' to 30. Child's method 'show' prints both its 'a' and parent's 'a' using 'super'. Instantiate the child class, call its 'show' method to display both values. Q.4 a) Apply the concept of abstraction in Java to design.		c) Apply aggregation in a Java program by designing two	[5]	002	
details such as emp_name and emp_id, and maintains a reference to a Bike object. Instantiate objects to illustrate the relationship, and display the employee's name and ID, along with the bike's name and registration number. d) Apply the 'super' keyword in Java to access parent class members within a child class. Parent initializes 'a' as 20, child class extends it, setting 'a' to 30. Child's method 'show' prints both its 'a' and parent's 'a' using 'super'. Instantiate the child class, call its 'show' method to display both values. Q.4 a) Apply the concept of abstraction in Java to design a virtual pet adoption center program. Create an abstract class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their area.	1	classes. Employee and Rike The Bile of the state of the s	[o]	CO3	Apply
reference to a Bike object. Instantiate objects to illustrate the relationship, and display the employee's name and ID, along with the bike's name and registration number. d) Apply the 'super' keyword in Java to access parent class members within a child class. Parent initializes 'a' as 20, child class extends it, setting 'a' to 30. Child's method 'show' prints both its 'a' and parent's 'a' using 'super'. Instantiate the child class, call its 'show' method to display both values. Q.4 a) Apply the concept of abstraction in Java to design a virtual pet adoption center program. Create an abstract class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their over		bike_name and regno while Employee close			
ID, along with the bike's name and registration number. d) Apply the 'super' keyword in Java to access parent class members within a child class. Parent initializes 'a' as 20, child class extends it, setting 'a' to 30. Child's method 'show' prints both its 'a' and parent's 'a' using 'super'. Instantiate the child class, call its 'show' method to display both values. Q.4 a) Apply the concept of abstraction in Java to design a virtual pet adoption center program. Create an abstract class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their own		details such as emp name and emp id and			
d) Apply the 'super' keyword in Java to access parent class members within a child class. Parent initializes 'a' as 20, child class extends it, setting 'a' to 30. Child's method 'show' prints both its 'a' and parent's 'a' using 'super'. Instantiate the child class, call its 'show' method to display both values. Q.4 a) Apply the concept of abstraction in Java to design a virtual pet adoption center program. Create an abstract class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their own		the relationship, and display the applement			
as 20, child class extends it, setting 'a' to 30. Child's method 'show' prints both its 'a' and parent's 'a' using 'super'. Instantiate the child class, call its 'show' method to display both values. Q.4 a) Apply the concept of abstraction in Java to design a virtual pet adoption center program. Create an abstract class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their own.		ID, along with the bike's name and registration number.			
method 'show' prints both its 'a' and parent's 'a' using 'super'. Instantiate the child class, call its 'show' method to display both values. Q.4 a) Apply the concept of abstraction in Java to design a virtual pet adoption center program. Create an abstract class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their over		d) Apply the 'super' keyword in Java to access parent			
ysuper'. Instantiate the child class, call its 'show' method to display both values. Q.4 a) Apply the concept of abstraction in Java to design a virtual pet adoption center program. Create an abstract class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their over		as 20, child class extends it, setting 'n' to 20, of the	[5]	СОЗ	Apply
Q.4 a) Apply the concept of abstraction in Java to design a virtual pet adoption center program. Create an abstract class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their average [5]		method show prints both its 'a' and parent's 'a'	i		
Q.4 a) Apply the concept of abstraction in Java to design a virtual pet adoption center program. Create an abstract class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their average.		super. Histarillate the child class, call its 'show' mothed			
class named Pet with a method speak(). Implement subclasses like Dog, Cat, and Parrot, extending Pet, each providing their own implementation of the speak() method to represent their unique sounds. b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their average [5]	$\overline{\bigcirc 4}$	to display both values.			
b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their average [5]	۷,٦	a) Apply the concept of abstraction in Java to design a	[5]	CO4	Apply
b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their average [5]		class named Pet with a method speak() Involved			
b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their course.		Subclasses like Dog, Cat, and Parrot extending Det			
b) Apply the concept of interface in Java to design a music player application. Create an interface named PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their course.		providing their own implementation of the great-of			,
PlayerControl with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their course.		method to represent their unique sounds.			
Player Control with functions for basic playback operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their courses.		b) Apply the concept of interface in I			
operations like play() and pause(). Implement classes such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their such		music player application. Create an interface	r=-3		
such as MP3Player and CDPlayer, both implementing the PlayerControl interface and offering their such		rayer control with functions for basic plants 1	[5]	CO4	Apply
the PlayerControl interface and offering their course		operations like play() and pause() Implement along -			
the PlayerControl interface and offering their own		such as MP3Player and CDPlayer both implement			
implementations for the playback controls.		the PlayerControl Interface and offering their own			
		implementations for the playback controls.			
c) Design a Java application for managing employee		c) Design a Java application for managing and			
records. Deline an Employee class with attributes and		records. Define an Employee class with attributes and	[6]	004	Annly
as name, age, and employee ID along with most		as name, age, and employee ID along with most	[9]	CO4	Apply
set and get these attributes. In the main matters		· · · · · · · · · · · · · · · · · · ·			
instantiate an Employee object, set its attributes with		set and get these attributes. In the main mathematical	,	1	
sample data, and display the employee's details.		instantiate an Employee object, set its attributes with		,	1
d)) Illustrate the concept of Abstract Class and abstract [5] CO4		set and get these attributes. In the main mathematical			İ
method by using Suitable java program. [5] CO4 Apply		instantiate an Employee object, set its attributes with sample data, and display the employee's details. d)) Illustrate the concept of Abstract Class and obstract	re1	204	