

I

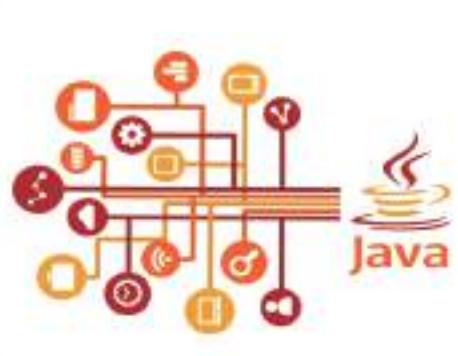
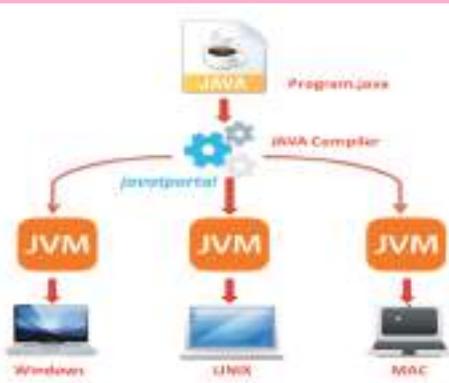
Name _____

Roll No. _____ Year 20 ____ 20 ____

Exam Seat No. _____

COMPUTER GROUP | SEMESTER - IV | DIPLOMA IN ENGINEERING AND TECHNOLOGY

A LABORATORY MANUAL
FOR
JAVA
PROGRAMMING
(22412)



MAHARASHTRA STATE BOARD OF TECHNICAL EDUCATION, MUMBAI
(Autonomous) (ISO 9001 : 2015) (ISO / IEC 27001 : 2013)

VISION

To ensure that the Diploma level Technical Education constantly matches the latest requirements of technology and industry and includes the all-round personal development of students including social concerns and to become globally competitive, technology led organization.

MISSION

To provide high quality technical and managerial manpower, information and consultancy services to the industry and community to enable the industry and community to face the changing technological and environmental challenges.

QUALITY POLICY

We, at MSBTE are committed to offer the best in class academic services to the students and institutes to enhance the delight of industry and society. This will be achieved through continual improvement in management practices adopted in the process of curriculum design, development, implementation, evaluation and monitoring system along with adequate faculty development programmes.

CORE VALUES

MSBTE believes in the followings:

- Education industry produces live products.
- Market requirements do not wait for curriculum changes.
- Question paper is the reflector of academic standards of educational organization.
- Well designed curriculum needs effective implementation too.
- Competency based curriculum is the backbone of need based program.
- Technical skills do need support of life skills.
- Best teachers are the national assets.
- Effective teaching learning process is impossible without learning resources.

**A Laboratory Manual
for**

**Java Programming
(22412)**

Semester-IV

(CO, CM, CW, IF)



**Maharashtra State
Board of Technical Education, Mumbai
(Autonomous) (ISO 9001:2015) (ISO/IEC 27001:2013)**



Maharashtra State Board of Technical Education,
(Autonomous) (ISO 9001 : 2015) (ISO/IEC 27001 : 2013)
4th Floor, Government Polytechnic Building, 49, Kherwadi,
Bandra (East), Mumbai - 400051.
(Printed on November 2018)



Maharashtra State Board of Technical Education

Certificate

This is to certify that Mr. / Ms.

Roll No.....of Fourth Semester of Diploma in
..... of Institute

.....
(Code.....) has attained predefined practical outcomes
(PROs) satisfactorily in course **Java Programming (22412)** for the
academic year 20.....to 20..... as prescribed in the curriculum.

Place

Enrollment No.....

Date:

Exam Seat No.

Course Teacher

Head of the Department

Principal



Preface

The primary focus of any engineering laboratory/field work in the technical education system is to develop the much-needed industry relevant competencies and skills. With this in view, MSBTE embarked on this innovative ‘I’ Scheme curricula for engineering Diploma programmes with outcome-based education as the focus and accordingly, relatively large amount of time is allotted for the practical work. This displays the great importance of laboratory work making each teacher, instructor and student to realize that every minute of the laboratory time need to be effectively utilized to develop these outcomes, rather than doing other mundane activities. Therefore, for the successful implementation of this outcome-based curriculum, every practical has been designed to serve as a ‘**vehicle**’ to develop this industry identified competency in every student. The practical skills are difficult to develop through ‘chalk and duster’ activity in the classroom situation. Accordingly, the ‘I’ scheme laboratory manual development team designed the practicals to **focus** on **outcomes**, rather than the traditional age-old practice of conducting practical’s to ‘verify the theory’ (which may become a byproduct along the way).

This laboratory manual is designed to help all stakeholders, especially the students, teachers and instructors to develop in the student the pre-determined outcomes. It is expected from each student that at least a day in advance, they have to thoroughly read the concerned practical procedure that they will do the next day and understand minimum theoretical background associated with the practical. Every practical in this manual begins by identifying the competency, industry relevant skills, course outcomes and practical outcomes which serve as a key focal point for doing the practical. Students will then become aware about the skills they will achieve through procedure shown there and necessary precautions to be taken, which will help them to apply in solving real-world problems in their professional life.

This manual also provides guidelines to teachers and instructors to effectively facilitate student-centered lab activities through each practical exercise by arranging and managing necessary resources in order that the students follow the procedures and precautions systematically ensuring the achievement of outcomes in the students.

Java is one of the most popular programming languages used to create Web applications and platforms. It is designed to allow developers to write code that would run on any machine, regardless of architecture or platform. Diploma pass out should be able to use Object Oriented Programming construct of java and will be able to create the applications by using object-oriented concepts. They should possess basic skills of programming syntax and naming conventions. This course is designed to develop these vital skills in them through lab-based activities.

Although all care has been taken to check for mistakes in this laboratory manual, yet it is impossible to claim perfection especially as this is the first edition. Any such errors and suggestions for improvement can be brought to our notice and are highly welcome.

Programme Outcomes (POs) to be achieved through Practicals of this Course:-

Following programme outcomes are expected to be achieved significantly out of the ten programme outcomes and Computer Engineering programme specific outcomes through the practicals of the course on **Java Programming**.

PO 1. Basic knowledge: Apply knowledge of basic mathematics, sciences and basic engineering to solve the broad-based Electronics related problems.

PO 2. Discipline knowledge: Apply Computer Programming knowledge to solve broad-based Electronics related problems.

PO 3. Experiments and practice: Plan to perform experiments and practices to use the results to solve broad-based Electronics related problems.

PO4. Engineering tools: Apply relevant Computer programming.
technologies and tools with an understanding of the limitations.

PO 8. Individual and teamwork: Function effectively as a leader and team member in diverse/ multidisciplinary teams.

PO 9. Communication: Communicate effectively in oral and written form.

Practical- Course Outcome matrix

Course Outcomes (COs)								
Sr. No.	Title of the Practical	CO a.	CO b.	CO c.	CO d.	CO e.	CO f.	
1*	Setup a Java Programming development environment by using: a) Command Prompt. (Classpath and path setup) b) Any IDE (Eclipse, Jcreator etc.).	✓	-	-	-	-	-	
2	Test the JDE setup by implementing a small program.	✓	-	-	-	-	-	
3*	Develop programs to demonstrate use of if statements and its different forms.	✓	-	-	-	-	-	
4*	Develop programs to demonstrate use of a) Switch - Case statement b) Conditional if (? :)	✓	-	-	-	-	-	
5*	Develop programs to demonstrate use of Looping Statement ‘for’	✓	-	-	-	-	-	
6*	Develop programs to demonstrate use of ‘while’, ‘do-while’	✓	-	-	-	-	-	
7	Develop programs for implementation of implicit type casting in Java. Part-I	✓	-	-	-	-	-	
8	Develop programs for implementation of implicit type casting in Java. Part-II	✓	-	-	-	-	-	
9*	Develop programs for implementation of explicit type conversion in Java.	✓	-	-	-	-	-	
10*	a) Develop programs for implementation of constructor b) Develop programs for implementation of multiple constructors in the class	✓	-	-	-	-	-	
11	Develop program for implementation of different functions of String Class. Part- I	✓	-	-	-	-	-	
12	Develop program for implementation of different functions of String Class. Part- II	✓	-	-	-	-	-	
13*	Develop program for implementation of Arrays in Java	✓	-	-	-	-	-	
14*	Develop program for implementation of Vectors in Java	✓	-	-	-	-	-	
15*	Develop a program for implementation of Wrapper Class to convert primitive into object.	✓	-	-	-	-	-	
16*	Develop a program for implementation of Wrapper Class to convert object into primitive.	✓	-	-	-	-	-	

17*	Develop a program which implements the concept of overriding.	-	√	-	-	-	-
18*	Develop a program for implementation of Single and multilevel inheritance.	-	√	-	-	-	-
19*	Develop a program for implementation multiple inheritance.	-	√	-	-	-	-
20*	Develop a program to import different classes in packages.	√	√	-	-	-	-
21*	Develop a program for implementation of multithreading operation Part-I	-	-	√	-	-	-
22	Develop a program for implementation of multithreading operation Part-II	-	-	√	-	-	-
23	Develop a program for implementation of try, catch block. Part-I	-	-	-	√	-	-
24	Develop a program for implementation of try, catch block. Part-II	-	-	-	√	-	-
25*	Develop a program for implementation of try, catch and finally block.	-	-	-	√	-	-
26*	Develop a program for implementation of throw, throws clause. Part-I	-	-	-	√	-	-
27*	Develop a program for implementation of throw, throws clause. Part-II	-	-	-	√	-	-
28*	Develop minimum two basic Applets. Display output with appletviewer and browser. a)Develop a program on basic applet. b)Develop a program using control loops in applets	-	-	-	-	√	-
29*	Write a program to create animated shape using graphics and applets. You may use following shapes: a) Lines and Rectangles. b) Circles and Ellipses. c) Arcs d)Polygons with fillpolygon method.	-	-	-	-	√	-
30*	Develop a program to draw following shapes, graphics and applets. a) Cone b)Cylinders c)Cube d) Square inside a circle e) Circle inside a square	-	-	-	-	√	-
31*	Develop a program for implementation of I/O stream classes.	-	-	-	-	-	√
32*	Develop a program for implementation of file stream classes.	-	-	-	-	-	√

List of Industry Relevant Skills

The following industry relevant skills or the competencies are expected to be developed in you by undertaking the practicals of this laboratory manual.

1. Develop Applications using Java.
2. Develop Applets by using Java.

Brief Guidelines to Teachers

Hints regarding strategies to be used:-

1. Teacher shall explain prior concepts to the students before starting each experiment.
2. For practical's requiring tools to be used, teacher should provide the demonstration of the practical emphasizing the skills, which the student should achieve.
3. Involve students in the activities during the conduct of each experiment.
4. Teachers should give opportunity to students for hands-on after the demonstration.
5. Assess the skill achievement of the students and COs of each unit.
6. Teacher is expected to share the skills and competencies to be developed in the students.
7. Teacher should ensure that the respective skills and competencies are developed in the students after the completion of the practical exercise.
8. Teacher may provide additional knowledge and skills to the students even though that may not be covered in the manual but are expected from the students by the industries.
9. Teacher may suggest the students to refer additional related literature of the reference books/websites/seminar proceedings etc.
10. During assessment teacher is expected to ask questions to the students to tap their knowledge and skill related to that practical.

Instructions for Students

Student shall read the points given below for understanding the theoretical concepts and practical applications.-

1. Students shall listen carefully the lecture given by teacher about importance of subject, learning structure, course outcomes.
2. Students shall organize the work in the group of two or three members and make a record of all observations.
3. Students shall understand the purpose of experiment and its practical implementation.
4. Students shall write the answers of the questions during practical.
5. Student should feel free to discuss any difficulty faced during the conduct of practical.
6. Students shall develop maintenance skills as expected by the industries.
7. Students shall attempt to develop related hands on skills and gain confidence.
8. Students shall refer technical magazines, websites related to the scope of the subjects and update their knowledge and skills.
9. Students shall develop self-learning techniques.
10. Students should develop habit to submit the write-ups on the scheduled dates and time.

Content Page
List of Practicals and Progressive Assessment Sheet

Sr. No .	Title of the practical	Page No.	Date of performance	Date of submission	Assessment marks (50)	Dated sign. of teacher	Remarks (if any)
1.	Setup a Java Programming development environment by using: a) Command Prompt. (Classpath and b) path setup) c) Any IDE (Eclipse, Jcreator etc.).	1					
2.	Test the JDE setup by implementing a small program.						
3.	Develop programs to demonstrate use of if statements and its different forms.	8					
4.	Develop programs to demonstrate use of a) Switch - Case statement b) Conditional if (?:)	16					
5.	Develop programs to demonstrate use of Looping Statement ‘for’	23					
6.	Develop programs to demonstrate use of ‘while’, ‘do-while’	29					
7.	Develop programs for implementation of implicit type casting in Java. Part-I	35					
8.	Develop programs for implementation of implicit type casting in Java. Part-II						
9.	Develop programs for implementation of explicit type conversion in Java.	41					
10.	a) Develop programs for implementation of constructor b) Develop programs for implementation of multiple constructors in the class	47					
11.	Develop program for implementation of different functions of String Class. Part- I	53					
12.	Develop program for implementation of different functions of String Class. Part- II						
13.	Develop program for implementation of Arrays in Java	60					

Sr. No.	Title of the practical	Page No.	Date of performance	Date of submission	Assessment marks (50)	Dated sign. of teacher	Remarks (if any)
14.	Develop program for implementation of Vectors in Java	65					
15.	Develop a program for implementation of Wrapper Class to convert primitive into object.	70					
16.	Develop a program for implementation of Wrapper Class to convert object into object.						
17.	Develop a program which implements the concept of overriding.	75					
18.	Develop a program for implementation of Single and multilevel inheritance.	81					
19.	Develop a program for implementation multiple inheritance.	88					
20.	Develop a program to import different classes in packages.	97					
21.	Develop a program for implementation of multithreading operation Part-I	104					
22.	Develop a program for implementation of multithreading operation Part-II						
23.	Develop a program for implementation of try, catch block. Part-I	113					
24.	Develop a program for implementation of try, catch block. Part-II						
25.	Develop a program for implementation of try, catch and finally block.						
26.	Develop a program for implementation of throw, throws clause. Part-I	121					
27.	Develop a program for implementation of throw, throws clause. Part-II						
28.	Develop minimum two basic Applets. Display output with applet viewer and browser. a) Develop a program on basic applet. b) Develop a program using control loops in applets	127					

Sr. No .	Title of the practical	Page No.	Date of performance	Date of submission	Assessment marks (50)	Dated sign. of teacher	Remarks (if any)	
29.	Write a program to create animated shape using graphics and applets. You may use following shapes: a) Lines and Rectangles. b) Circles and Ellipses. c) Arcs d) Polygons with fill polygon method.	135						
30.	Develop a program to draw following shapes, graphics and applets. a) Cone b) Cylinders c) Cube d) Square inside a circle e) Circle inside a square	142						
31.	Develop a program for implementation of I/O stream classes.	148						
32.	Develop a program for implementation of file stream classes.							
Total Marks								
Total Marks(Scaled to 50 Marks)								

- To be transferred to Proforma of CIAAN-2017

Practical No. 1 and 2: Setup a Java programming development environment and test using small program.

I. Practical Significance:

Java is the popular platform, which is used to develop various applications for the systems as well as embedded devices like mobile, laptops, tablets and many more. It is an object-oriented programming language. Students will be able to setup Java environment for executing Java programs, using command prompt or using different IDEs like Eclipse, JCreator and test the setup using small java program.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Teamwork:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Set up Java Environment for executing Java programs.
2. Execute simple program by setting path variable .

Setup a java programming development environment.

1. Using Command prompt
2. Using IDEs.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

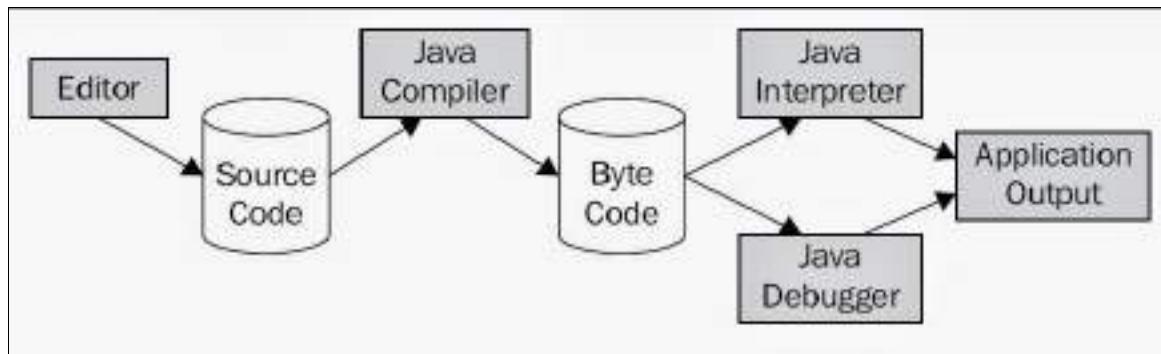
Setup a java programming development environment.

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Java language is compiled and interpreted.

**Fig. 1 Java Program Execution****1. Procedure**

Installation for Java Software:

(On a PC loaded with windows OS – 2000/2003/2007 onwards with notepad)

1. Download JDK (jdk 1.4.0 onwards)
Visit the
<https://www.oracle.com/technetwork/java/javase/downloads/index.html>
Download the windows version to suitable folder.
2. Double click the setup file.
3. Follow onscreen instruction.
4. When the setup is done, the complete screen appears, click on the ‘Finish’ Button.
This completes the installation of JDK. To ensure the JDK installation / to determine the java version, type the following command at the MS Dos prompt:
<system prompt> java –version
It should show the output similar to following

```

C:\Program Files\IBM\Java60\bin>java -version
java version "1.6.0"
Java(TM) SE Runtime Environment (build pwi3260sr5-20090529_04(SR5))
IBM J9 VM (build 2.4, J2RE 1.6.0 IBM J9 2.4 Windows XP x86-32 jvmwi3260sr5-20090519_35743 (JIT enabled, AOT enabled)
J9VM - 20090519_035743_1HdSMr
JIT - r9_20090518_2017
GC - 20090417_AA
JCL - 20090529_01

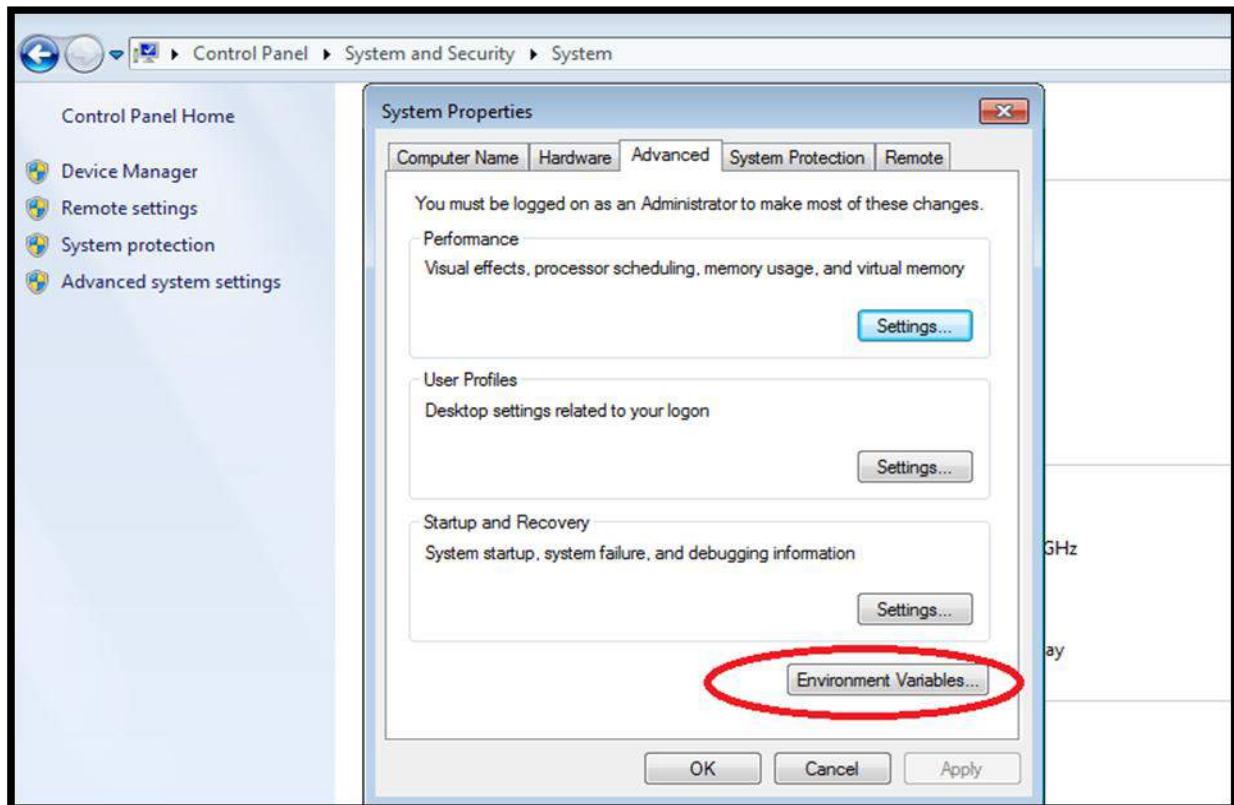
C:\Program Files\IBM\Java60\bin>

```

Fig. 2 Java Version

If not then set ‘path’ environment variable

1. Go to start -> control panel -> system
2. System properties dialogbox will appear.
3. Select ‘advanced tab’ -> environment variables.
4. In the system variable list, select path and click ‘edit’
5. Edit the system variable dialogbox appears. In the variable value field , append the path to the JDK bin directory (generally ‘c:\program files\java\jdk1.6.0\bin’) at the end. Use semicolon to separate the path of bin directory from the rest of values already available. Click ‘Ok’.
6. Similarly set ‘classpath’ environment variable

**Fig. 3 Environment variable**

Note: Follow the similar instructions for other platforms (say Unix, Linux, Mac) with appropriate jdk download.

2. Using an Eclipse for Java

Eclipse (www.eclipse.org) is an *open-source* Integrated Development Environment (IDE) supported by IBM. Eclipse is popular for Java application development (Java SE and Java EE) and Android apps.

Installing Eclipse 4.7.2 (Oxygen 2) for Java Developers

To use Eclipse for Java programming, you need to first install Java Development Kit (JDK).

1. Download Eclipse from <https://www.eclipse.org/downloads>. Under "Get Eclipse Oxygen" ⇒ Click "Download Packages".
2. To install Eclipse, unzip the downloaded file into a directory (e.g., "d:\myproject").

3. Testing setup using small program:

Steps for editing and executing java program:

Using an editor (e.g. Notepad)

1. Open notepad
2. Write the program (called java source code) in notepad
3. Save the file as 'filename.java' in some directory. The filename must be same as the classname containing main() method.
4. Open MS-Dos prompt.
5. Change the directory containing to the one containing the program.
6. Compile the program by using the command **javac<filename.java>**

7. Execute/ Run the program by using the command **java <filename>**

4. Using Eclipse

1. Launch Eclipse by running "eclipse.exe" from the Eclipse installed directory.
2. Choose an appropriate directory for your *workspace*
3. To create a new Java project using "File" menu \Rightarrow "New" \Rightarrow "Java project"
4. In "JRE", select "Use default JRE (currently 'JDK9.0.x')". But make sure that your JDK is 1.8 and above.
5. In "Project Layout" menu, select "Use project folder as root for sources and class files".
6. Push "Finish" button.
7. In the "Package Explorer" (left pane) \Rightarrow Right-click on "FirstProject" \Rightarrow New \Rightarrow Class.
8. Write a program
9. Compile and execute program.
10. Observe output on the console panel.

Sample program:

```
Class HelloWorld
{
    public static void main(String args[])
    {
        System.out.println("Welcome to Hello World program");
    }
}
```

5. Output:

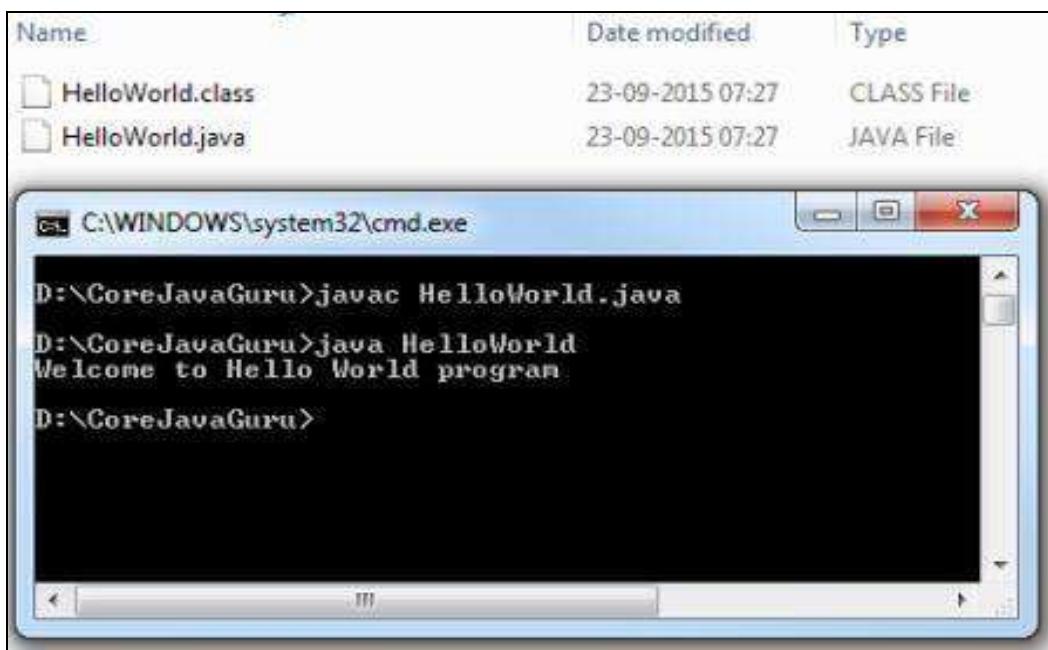


Fig 4. Output of the Program

VIII. Resources required

Sr. No.	Name of Resource	Suggested Broad Specification	Quantity	Remark
1	Hardware: Computer System	Computer (i3-i5 preferable), RAM minimum 2 GB and onwards	As per batch size	For all Experiments
2	Operating system	Windows / Linux		
3	Software	jdk1.8.0 or above.		

IX. Resources used

S. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1	Computer System with broad specifications			
2	Software			
3	Any other resource used			

X. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Write installation directory path of your directory?
2. Write value of path environment variable?
3. List folders created after installation.
4. Main method is declared as static. Justify.
5. Program is named with class containing main method. Justify.

(Space for answer)

XI. Exercise

- Write the options provided by following JDK tools along with their use
1. java 2. javac 3. javadoc
 2. List different versions of JDK.
 3. Test the setup using similar programs.

XII. References/ Suggestions for Further Reading

1. <https://www.javatpoint.com/internal-details-of-jvm>
2. <https://www.geeksforgeeks.org/jvm-works-jvm-architecture/>
3. <https://www.youtube.com/watch?v=evoLlsLFn10>
4. <https://www.youtube.com/watch?v=e7DXhdCsSnw>

XIII. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission of report	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related(35)	Product Related(15)	Total (50)	

Practical No. 3: Develop programs to demonstrate use of if statements and its different forms.

I. Practical Significance:

In computer science, conditional statements, expressions and constructs are perform different computations or actions depending on whether boolean condition evaluates to true or false. Students will be able to use various forms of if statements to check the condition.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Write a program to use simple if statements to check conditions
2. Develop a program to use different forms of if to check multiple conditions.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop programs to demonstrate use of if statements and its different forms.

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Decision making in Java programming

Control statements are used to control the flow of execution of program based on certain conditions. These are used to cause the flow of execution to advance and branch based on changes to the state of program.

Java Selection Statements:

1. if
2. if-else
3. nested-if
4. if-else-if ladder

1. **if:** if statement is simple decision-making statement. It is used to decide whether a certain statement or block of statements will be executed or not. i.e. if a certain condition is true then the block will be executed otherwise not.

Syntax:

```
if(condition)
{
    // Statement to execute if the condition is true;
}
```

2. **if-else:** The if statement alone tells us that if a condition is true it will execute a block of statements and if the condition false, else block will be executed.

Syntax:

```
if(condition)
{
    // Statement to execute if the condition is true;
}
else
{
    // Statement to execute if the condition is false;
}
```

3. **nested-if:** A nested if is an if statement that is the target of another if or else. Nested if statements means an if statement inside an if statement.

```
if(condition1)
{
    // execute when condition1 is true.
if(condition2)
{
    // execute when condition2 is true.
}
}
```

4. **if-else-if ladder:** A user can decide among multiple options. The if statements are executed from top down. When one of condition is true, the statement associated with that if is executed, and the rest of the ladder is bypassed. If none of the conditions is true, then the final else statement will be executed.

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Write any program to check multiple conditions using if statement.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. List operators used in if conditional statement.
2. In if-else construct which part will be executed if condition is true.
3. State the condition when the else part will be executed with example.

4. Which of the following operator is used in if:
a. Assignment operator (=) b. comparison operator (==)

(Space for answer)

XIII. Exercise:

1. Write output of code in the given space.

Sr. No.	Program Code	Output
1.	<pre>public class NestedIfExample { public static void main(String args[]){ int num=70; if(num< 100){ System.out.println("number is less than 100"); if(num> 50){ System.out.println("number is greater than 50"); } } } }</pre>	
2.	<pre>class IfStatement { public static void main(String[] args) { int number = 10; if (number > 0) { System.out.println("Number is positive."); } System.out.println("This statement is always executed."); } }</pre>	

2. Write a program to make the use of logical operators.
 3. Write a program to check no is even or odd.

(Space for Answer)

XIV. References/ Suggestions for Further Reading

1. <https://www.youtube.com/watch?v=uNiryYp2vSg> 2
2. <https://www.youtube.com/watch?v=KTvZkBfFR8U>
3. <https://www.javatpoint.com/java-if-else>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission of report	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related(35)	Product Related(15)	Total (50)	

Practical No. 4: Develop programs to demonstrate use of switch – case statement and conditional if (?:)

I. Practical Significance:

Java uses control statements to control the flow of execution of program based on certain conditions. These are used to cause the flow of execution to advance and branch based on condition. Students will be able to use switch-case to check the multiple conditions.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Write a program to use switch-case to check multiple conditions.
2. Develop a program to check condition in one line.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop programs to demonstrate use of switch - case statement and conditional if (?:)

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

1. Decision making using Switch-case:

Syntax:

```
switch(expression)
{
    case value1 :
        // Statements
        break; // break is optional
```

```
case value2 :  
    // Statements  
    break; // break is optional  
  
.  
  
.  
  
case valueN :  
    // Statements  
    break; // break is optional  
  
default :  
    // Statements  
}
```

2. Conditional if (ternary operator):
Syntax:
result = testStatement ? value1 : value2;

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Write any program using switch-case statement.

XI. Result (Output of Code):

.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. What will happen if break is not written for a case in switch case?
 2. When default case is executed?
 3. List datatypes allowed in switch expression?
 4. Write a program to make use of ternary operator.

(Space for Answer)

XIII. Exercise:

- Write Error/output of code in the given space.

Sr. No.	Program Code	Error/Output
1.	<pre>public class SwitchCaseExample1 { public static void main(String args[]){ int num=2; switch(num+2) { case 1: System.out.println("Case1: Value is: "+num); case 2: System.out.println("Case2: Value is: "+num); case 3: System.out.println("Case3: Value is: "+num); default: System.out.println("Default: Value is: "+num); } } }</pre>	
2.	<pre>public class Program { public static void main(String[] args) { int value = 100; switch (value) { case 100: System.out.println(true); break; case 100: System.out.println(true); break; } } }</pre>	

- Write any program to check switch-case statement using character datatype.

(Space for Answer)

XIV. References/ Suggestions for Further Reading

1. <https://www.geeksforgeeks.org/switch-statement-in-java/>
2. <https://www.guru99.com/switch-java.html>
3. <https://www.youtube.com/watch?v=g5Kphflexzg>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission of report	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related(35)	Product Related (15)	Total (50)	

Practical No. 5: Develop programs to demonstrate use of looping statement ‘for’.

I. Practical Significance:

A for loop is used to execute a block of code several times. Students will be able to use for loop to replace the repetition of statements.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Develop a program to using for loop

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop programs to demonstrate use of looping statement ‘for’

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Syntax:

```
for (initialization condition; testing condition; increment/decrement)
{
    statement(s);
}
```

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Develop a program to print command line argument using for loop.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. When for loop will be terminated?
 2. Can we write a for loop without initialization? If yes, give example.
 3. Write a for loop to increment index variable by 2 in each iteration.
 4. When for loop will be executed infinitely?

(Space for answer)

XIII. Exercise:

1. Write any program using if condition with for loop.
 2. Write any program to display pyramids of stars/patterns using increment/decrement

(Space for Answer)

XIV. References/ Suggestions for Further Reading

1. <https://www.sitesbay.com/java/java-looping-statement>
2. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/for.html>
3. <https://www.youtube.com/watch?v=0ll7vm1GQYE>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related(35)	Product Related(15)	Total(50)	

Practical No. 6: Develop programs to demonstrate use of ‘while’, ‘do-while’

I. Practical Significance:

Loop is used in programming to repeat a specific block of code until certain condition is true. Students will be able to use while and do-while loop to replace the repetition of statements.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Develop a program to using while and do-while loop.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop programs to demonstrate use of ‘while’, ‘do-while’

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

1. While loop:

Syntax:

```
while(condition)
{
    statement(s);
}
```

2. do-while loop:

Syntax:

```
do {
    // Statements
} while (Boolean_expression);
```

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Develop a program to use logical operators in do-while loop.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. State difference between while and do-while loop.
 2. In do-while loop termination condition is checked at _____(beginning/end)
 3. How many times do-while loop will be executed if condition is false?

(Space for answer)

XIII. Exercise:

- #### **Exercise:**

Sr. No.	Program Code	Error/Output
1.	<pre data-bbox="336 1205 865 1540"> class DoWhileBasics { public static void main(String args[]) { int a=1; do { System.out.println(a); a=a+1; // or a++; } while(a<=10); } }</pre>	
2.	<pre data-bbox="336 1540 865 1827"> class Test { public static void main(String[] args) { while(true) { System.out.print(1); do { System.out.print(2); } while (false); } } }</pre>	

2. Write a program to display number 1 to 50 using do-while loop.

XIV. References/ Suggestions for Further Reading

1. <https://www.codesdope.com/c-loop-and-loop/>
 2. <https://www.youtube.com/watch?v=llX6cLed73o>
 3. <https://www.journaldev.com/16536/java-do-while-loop>
 4. <https://beginnersbook.com/2015/03/do-while-loop-in-java-with-example/>

(Space for answer)

XV. Assessment Scheme

Performance Indicators		Weightage
Process related (35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 7 and 8: Develop programs for implementation of implicit type casting in Java, Part –I and Part – II.

I. Practical Significance:

Assigning a value of one type to a variable of different type is known as Type Casting. When you assign value of one data type to another, the two types might/ might not be compatible with each other. Students will be able to understand implicit type conversion between data types.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

Develop Applications using Java.

IV. Relevant Course Outcome(s)

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Develop a program to show automatic conversion between various compatible data types.

V. Practical Outcome (PrOs)

Develop programs for implementation of implicit type casting in Java.

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

1. Widening or automatic type conversion: Possible when two types are compatible and target type is greater than source type.
2. Narrowing may result in loss of information.

Following table shows the casts that result in a loss of information.

Sr. No.	From	To
1.	byte	short, char, int, long, float, double
2.	short	int, long, float, double
3.	char	int, long, float, double
4.	long	float, double
5.	float	double

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Develop a program to show the use of implicit typecasting.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. List different data types according to storage capacity.
 2. State need of typecasting.
 3. State the data types to which boolean datatype is implicitly casted.
 4. Write two examples of implicit type casting.

(Space for answer)

XIII. Exercise:

1. Write Error/output of code in the given space.

Sr. No.	Program Code	Error/Output
1.	<pre>class Test{ public static void main(String[] args) { int i = 100; long l = i; float f = l; System.out.println("Int value "+i); System.out.println("Long value "+l); System.out.println("Float value "+f); } }</pre>	
2.	<pre>public class Test{ public static void main(String[] argv) { char ch = 'c'; int num = 88; ch = num; } }</pre>	

3. Write a program to implicitly typecast lower range data type to larger storage size datatype.

XIII. References/ Suggestions for Further Reading

1. <http://www.javainterviewpoint.com/type-casting-java-implicit-explicit-casting/>
2. <https://www.dyclassroom.com/java/java-type-casting>
3. <https://www.studytonight.com/java/type-casting-in-java>

XIV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total(50)	

Practical No. 9: Develop programs for implementation of explicit type conversion in Java.

I. Practical Significance:

Assigning a value of one datatype to another datatype is known as typecasting. When larger type of data should be assigned to smaller datatype explicit typecasting is required. Students will be able to implement assignment of explicit typecasting.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Develop a program to show use of explicit type casting.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop programs for implementation of explicit type conversion in Java.

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Narrowing or Explicit Conversion

If we want to assign a value of larger data type to a smaller data type we perform type casting explicitly or narrowing.

- This is useful for incompatible data types.
- Here, target-type specifies the desired type to convert the specified value to.

Syntax:

dataType variableName = (dataType) variableToConvert;

Example:

```
float a =5.2;  
int b = (float) a;
```

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Develop a program to show the use of explicit type casting.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. What is casting?
 2. What is difference between implicit and explicit type casting?
 3. What is narrowing?

(Space for answer)

XIII. Exercise:

1. Write Error/output of code in the given space.

Sr. No.	Program Code	Error/Output
1.	<pre>class Test{ public static void main(String[] args) { double d = 100.04; long l = (long)d; int i = (int)l; System.out.println("Double value "+d); System.out.println("Long value "+l); System.out.println("Int value "+i); } }</pre>	
2.	<pre>class Test{ public static void main(String args[]) { byte b = 50; b = (byte)(b * 2); System.out.println(b); } }</pre>	

```
3. class Test{
    public static void main(String args[])
    {
        byte a= 4;
        char b = 'z';
        short c = 102;
        int i = 5000;
        float f = 5.7f;
        double d = .124;
        double result = (f * a) + (i / b) - (d * c);
        System.out.println("result = " + result);
    }
}
```

2. Write a program to convert variable of basic datatypes and shows result of explicit typecasting.

(Space for Answer)

XIV. References/ Suggestions for Further Reading

1. <http://www.javainterviewpoint.com/type-casting-java-implicit-explicit-casting/>
2. <https://www.dyclassroom.com/java/java-type-casting>
3. <https://www.studytonight.com/java/type-casting-in-java>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related(35)	Product Related(15)	Total(50)	

Practical No. 10: Develop program for implementation of constructor and multiple constructors.

I. Practical Significance:

Constructor in Java or any programming language is a special type of method that is used to initialize the object. Different types of constructors are used to initialize the object in different way. The student will be able to use different types of constructor for creating the object.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitation
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Write a program to initialized objects using constructor.
2. Develop a program to pass message among objects.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop programs for implementation of single constructor and multiple constructors.

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Types of constructors

1. Default constructor (No-argument constructor)-
General Syntax –
`class_name()
{
 Statements for initialize the data members.
}`
2. Parameterized constructor
General Syntax –
`class_name(parameter list)`

- {
Statements for initialize the data members.
}
3. Copy Constructor
General Syntax –
class_name (classname reference)
{
Statements for initialize the data members.
}

VIII. Resources required(Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Demonstrate use of at least two types of constructors.

XI. Result (Output of Code):

.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Does constructor return a value?
 2. Specify the situation when the default constructor is provided by the system.
 3. Specify the situation when the default constructor is explicitly defined in the class.
 4. How constructor overloading can be done?

(Space for answer)

XIII. Exercise:

1. Write output of code in the given space.

Sr. No.	Program Code	Output
1.	<pre>class T { int t; } class Main { public static void main(String args[]) { T t1 = new T(); System.out.println(t1.t); } }</pre>	

2. Modify the following program to execute without error. State which constructors are used in the program.

```
class Point
{
    int m_x, m_y;
    public Point(int x, int y)
    {
        m_x = x; m_y = y; }
    public static void main(String args[])
    {
        Point p1 = new Point();
        Point p = new Point(2,3);
        System.out.println("X"+p.m_x);
        System.out.println("Y"+p.m_y);
        System.out.println("X"+p1.m_x);
        System.out.println("Y"+p1.m_y);
    }
}
```

3. Write a program to implement different types of constructors to perform addition of complex numbers.

(Space for Answer)

.....

.....

.....

.....

.....

.....

XIV. References/ Suggestions for Further Reading

1. <https://www.youtube.com/watch?v=lrYghXs9EEU>
2. <https://freevideolectures.com/course/2513/java-programming/17>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related (35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related(35)	Product Related(15)	Total(50)	

Practical No. 11 and 12: Develop program for implementation of different functions of String Class, Part – I and Part – II.

I. Practical Significance:

String is a sequence of characters. Java String is a powerful concept. Students will be able to perform various operations on String object using different methods of String class.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Write a program to initialized objects using constructor.
2. Develop a program to pass message among objects.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop program for implementation of different functions of String Class.

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

String class methods:

Sr. No.	Syntax	Task Performed
1.	public char charAt(int index)	Returns character value at specified index
2.	public int compareTo(String anotherString)	Compares two strings and returns int
3.	public boolean equals(Object anotherObject)	Compares two strings.
4.	public boolean equalsIgnoreCase (String str)	Compares two strings, ignoring cases.

5.	<code>public int length()</code>	Returns length of string
6.	<code>public String replace(char oldChar, char newChar)</code>	Returns a string replacing all the old char or CharSequence to new char or CharSequence.
7.	<code>public boolean startsWith(String prefix)</code>	Checks if this string starts with given prefix.
8.	<code>public boolean endsWith(String suffix)</code>	Checks if this string ends with given suffix.
9.	<code>int indexOf(int ch)</code>	A method returns index of given character value or substring.
10.	<code>String substring(int startIndex)</code>	Returns new string that is substring of this string
11.	<code>int lastIndexOf(int ch)</code>	A method returns last occurrence of given character value or substring.

String Buffer class methods:

Sr. No.	Syntax	Task Performed
1.	<code>StringBuffer append (StringBuffer sb)</code>	Appends specified StringBuffer with this StringBuffer
2.	<code>StringBuffer insert (int offset, String str)</code>	This method inserts a string str at position mentioned by offset.
3.	<code>void setLength(int newlength)</code>	Sets the length of the character sequence
4.	<code>void setCharAt(int index, char ch)</code>	The character at specified index of this StringBuffer is set to ch.
5.	<code>StringBuffer reverse()</code>	Reverse the character sequence in this StringBuffer.

VIII. Resources required(Additional)**Nil****IX. Resources used (Additional)**

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Write a program to show the use of all methods of String class.

XI. Result (Output of Code):

.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. List different constructors of String class along with syntax
 2. List different constructors of StringBuffer class along with syntax.
 3. State whether String is primitive datatype or class in Java? State the package.
 4. What is difference between == , equals() and compareTo() method?

(Space for answer)

XIII. Exercise:

1. Write output of code in the given space.

Sr. No.	Program Code	Output
1.	<pre>class String_demo{ public static void main(String args[]) { char chars[] = {'a','b','c'}; String s = new String(chars); System.out.println(s); } }</pre>	
2.	<pre>class Output{ public static void main(String args[]) { String s1 = "Hello I love Java"; String s2 = new String(s1); System.out.println((s1==s2) + " " + s1.equals(s2)); } }</pre>	

2. Write a program to implement all methods of StringBuffer class.

(Space for Answer)

.....
.....

XIV. References/ Suggestions for Further Reading

1. https://www.tutorialspoint.com/java/java_strings.htm
2. <https://beginnersbook.com/2013/12/java-strings/>
3. <http://nptel.ac.in/courses/106106145/29>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 13: Develop program for implementation of Arrays in Java

I. Practical Significance:

Arrays store homogeneous data i.e same type of data in consecutive memory locations which will help to fetch data in constant access time. Students will be able to use array to refer multiple values with same name.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Teamwork:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Develop a program to perform various operations using array.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop program for implementation of Arrays in Java

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

1. Creating an array:

```
dataType[] arrayRefVar;  
or  
dataType arrayRefVar[];  
arrayRefVar = new type [size];
```

2. Array of Objects:

```
Class_name array_name = new class_name[size];
```

3. Types of arrays:

1. One Dimensional

Example:

int [] intArray = new int[20];

2. Multi Dimensional

Example:

int[][] intArray = new int[10][20]; //a 2D array or matrix

int[][][] intArray = new int[10][20][10]; //a 3D array

VIII. Resources required(Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Write a program to implement multidimensional array.

XI. Result (Output of Code):

.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. What is use of new operator in defining an array?
 2. In 2D array which dimension is optional at the declaration of array?
 3. Is it possible to change size of array once allocated?
 4. State the situation where Index Out of Bounds Exception will be generated.

(Space for answer)

XIII. Exercise:

1. Write output/error of code in the given space.

Sr. No.	Program Code	Output/Error
1.	State line no and error. <pre data-bbox="381 804 941 980">class Test2 { public static void main(String[] args) { int a[] = new int[5]; // line 1 int[] arr = new int[]; // line 2 } }</pre>	
2.	<pre data-bbox="381 993 921 1003">class Test5 { public static void main(String[] args) { int arr[] = new int[5]; System.out.println(arr); System.out.println(arr[0]); } }</pre>	

3. Write a program to display array elements using for-each loop.

(Space for answer)

.....

.....

.....

.....

.....

XIV. References/ Suggestions for Further Reading

1. <https://www.javatpoint.com/array-in-java>
2. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html>
3. <https://www.youtube.com/watch?v=okHL1h5rhNM>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 14: Develop program for implementation of Vectors in Java

I. Practical Significance:

Vector implements a dynamic array. Vector hold different number of objects. Students will be able to use Vectors in the program efficiently.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Develop a program to perform various operations on Vector using different methods.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop program for implementation of Vectors in Java

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Vector class methods:

Sr. No.	Syntax	Task Performed
1.	void addElement(Object ob)	Adds the specified component to the end of this vector increasing its size by one
2.	int capacity()	Returns the current capacity of this vector
3.	boolean contains(Object elem)	Tests if the specified object is a component in this vector.
4.	void clear()	Removes all the elements from this vector
5.	Object elementAt(int index)	Returns the component at specified index

6.	Enumeration elements()	Returns enumeration of components of this vector
7.	Object firstElement()	Returns first component of this vector
8.	Object lastElement()	Returns last component of this vector
9.	int indexOf(Object elem)	Searches for the first occurrence of given argument.
10.	void insertElementAt(Object obj, int index)	Inserts specified object as a component at the specified index position.
11.	void removeElementAt(int index)	Removes the element at specified position in the vector
12.	boolean removeElement(Object obj)	Removes first occurrence of the argument from the vector
13.	int size()	Returns number of components in the vector
14.	void copyInto(Object[] array)	Copies the components of vector into specified array.

VIII. Resources required(Additional)**Nil****IX. Resources used (Additional)**

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Write a program to insert different elements in the Vector and display them.

XI. Result (Output of Code):

.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. State difference between size() and capacity() method of Vector class.
 2. Differentiate between addElement() and insertElementAt() methods of Vector class.
 3. Differentiate between array and Vector.

(Space for answer)

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

XIII. Exercise:

Sr. No.	Program Code	Output
1.	<pre>import java.util.*; class demo1 { public static void main(String[] args) { Vector v = new Vector(20); System.out.println(v.capacity()); System.out.println(v.size()); } }</pre>	

1. Write a program to use different methods of Vector class.

(Space for answer)

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

.....

.....

.....

.....

.....

.....

.....

XIV. References/ Suggestions for Further Reading

1. <https://www.javatpoint.com/array-in-java>
2. <https://docs.oracle.com/javase/tutorial/java/nutsandbolts/arrays.html>
3. <https://www.youtube.com/watch?v=okHL1h5rhNM>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1	Logic formation	30%
2	Debugging ability	30%
3	Follow ethical practices	10%
Product related (15 Marks)		30%
4	Expected output	10%
5	Timely Submission	10%
6	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 15 and 16: Develop a program for implementation of Wrapper Class to convert primitive into object and object into primitive.

I. Practical Significance:

Wrapper classes are used to convert primitive data type into an object. The primitive data types are not objects. They are predefined in the language itself. Student should be able to use different wrapper classes and their methods.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and Team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java”.

The practical is expected to develop the following skills:

1. Develop a program to create object of primitive data types and use them.

IV. Relevant Course Outcome(s)

Develop programs using Object Oriented methodology in Java.

V. Practical Outcome (PrOs)

Develop a program for implementation of Wrapper Class to convert primitive into object.

VI. Relevant Affective domain related Outcome(s)

1. Follow safety practices.
2. Practice good housekeeping
3. Demonstrate working as a leader/ a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Integer class methods:

Sr. No.	Method	Purpose
1.	parseInt(s)	returns a signed decimal integer value equivalent to string s
2.	toString(i)	returns a new String object representing the integer i
3	byteValue()	returns the value of this Integer as a byte
4	doubleValue()	returns the value of this Integer as a double
5	floatValue()	returns the value of this Integer as a float

6	<code>intValue()</code>	returns the value of this Integer as an int
7	<code>shortValue()</code>	returns the value of this Integer as a short
8	<code>longValue()</code>	returns the value of this Integer as a long
9	<code>int compareTo(int i)</code>	Compares the numerical value of the invoking object with that of i. Returns 0 if the values are equal. Returns a negative value if the invoking object has a lower value. Returns a positive value if the invoking object has a greater value.
10	<code>static int compare(int num1, int num2)</code>	Compares the values of num1 and num2. Returns 0 if the values are equal. Returns a negative value if num1 is less than num2. Returns a positive value if num1 is greater than num2.
11	<code>boolean equals(Object intObj)</code>	Returns true if the invoking Integer object is equivalent to intObj. Otherwise, it returns false.

Similar Wrapper class methods are available for Float, Short, Long and Double class.

VIII. Resources required(Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Write a program to show the use of Integer Wrapper class methods.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Write a different ways to create object of the any primitive datatype.
 2. Write methods of Number class to convert object into primitive datatypes.
 3. List all Wrapper classes in Java.

(Space for answer)

XIII. Exercise:

- EXERCISE:**

 1. Write a program to convert String value into Integer Wrapper class object.
 2. Write a program to make use of Character Wrapper class methods.
 3. Write a program to convert Integer object value into primitive datatype byte, short and double value.

(Space for answer)

.....

XIV. References/ Suggestions for Further Reading

1. <https://www.youtube.com/watch?v=IM3c6eI6lO8>
2. https://www.tutorialspoint.com/java/java_numbers.htm

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total(50)	

Practical No. 17: Develop a program which implements the concept of overriding.

I. Practical Significance:

Method overriding is used to change definition of method in subclass. Runtime Polymorphism is achieved through Method Overriding. super keyword is used to call the parent class method or constructor.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The practical is expected to develop the following skills:

1. Develop a program using super and sub class to override the methods.
2. Develop a program using super keyword to override the methods to achieve the runtime polymorphism.

IV. Relevant Course Outcome(s)

Apply concept of inheritance for code reusability.

V. Practical Outcome (PrOs)

Develop a program which implements the concept of overriding.

VI. Relevant Affective domain related Outcome(s)

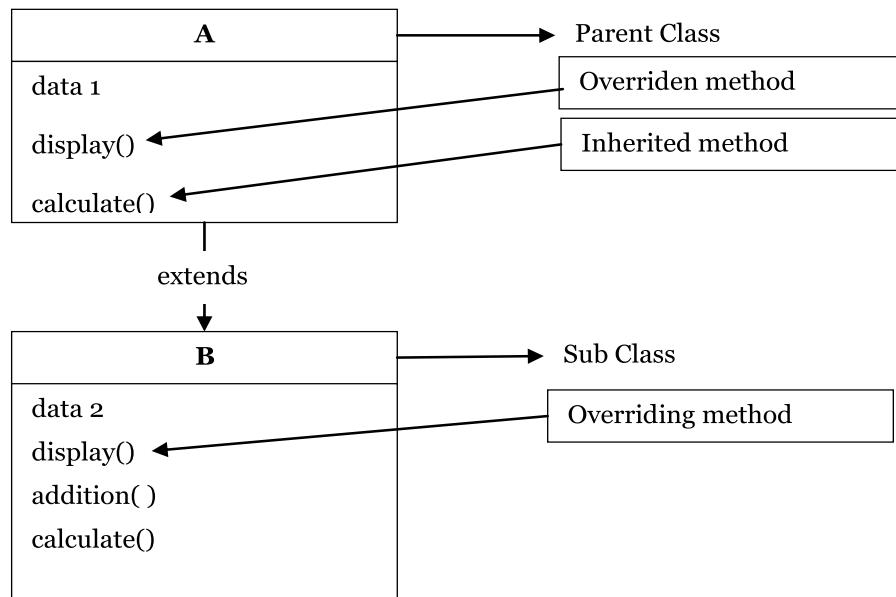
1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Method Overriding in Java

Declaring a method in a subclass (child class) which is already existing in the parent class refers to method overriding in java.

The exact implementation in the subclass (child class) overrides (replaces) the implementation in the superclass by providing a method that has same name, same parameters or signature and same return type as that of method in the super(parent) class.

**Fig. 5 Overriding of Methods****VIII. Resources required (Additional)****Nil****IX. Resources used (Additional)**

Sr. No.	Name of Resource	Broad Specification	Quantity	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Demonstrate the use of Overriding method display() using Super and Sub classes

Note: Attach the code at the end.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. State the difference between method overloading and method overriding.
 2. Method Overriding is an example of(Compile Time Polymorphism/Run Time Polymorphism)
 3. Write the rules of method overriding.
 4. Write the use of super keyword in method overriding.
(Space for answer)
-
.....
.....
.....
.....
.....

XIII. Exercise:

1. Develop a program to display the rate of interest of banks by method overriding method.
 2. Write the output of the following:

```
Write the output of the following:  
class MyBaseClass  
{  
    protected void disp()  
    {  
        System.out.println("Parent class method");  
    }  
}  
class MyChildClass extends MyBaseClass  
{  
    public void disp()  
    {  
        System.out.println("Child class method");  
    }  
    public static void main (String args[])  
    {  
        MyChildClass obj = new MyChildClass();  
        obj.disp();  
    }  
}
```

3. Develop a program to extend ‘dog’ from ‘animal’ to override ‘move()’ method using super keyword.

(Space for answer)

-

- XIV. References/ Suggestions for Further Reading**
1. <https://www.javatpoint.com/method-overriding-in-java>
 2. https://www.tutorialspoint.com/java/java_overriding.htm
 3. Programming with Java by E. Balagurusamy
 4. <https://beginnersbook.com/2014/01/method-overriding-in-java-with-example/>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 18: Develop a program for implementation of Single and Multilevel inheritance.

I. Practical Significance:

Inheritance helps to reuse the existing class properties to derive a new class with additional properties. It helps to reduce the memory space, time, frustration and increases the reliability of code. Different types of Inheritance are used to extend the classes in different ways. The student will be able to use different types of inheritance.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The practical is expected to develop the following skills:

1. Write simple programs to extend a subclass from super class
2. Develop programs to extend multipath inheritance.

IV. Relevant Course Outcome(s)

Apply concept of inheritance for code reusability.

V. Practical Outcome (PrOs)

Develop a program for implementation of Single and Multilevel inheritance.

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Inheritance:

The process of deriving a **new class** from an **old class** is called as **Inheritance**.

Types of Inheritance:

1. Single Inheritance
2. Multiple Inheritance
3. Hierarchical Inheritance
4. Multilevel Inheritance

Java does not directly implement multiple inheritance, it is implemented using a secondary inheritance path in the form of **interfaces**.

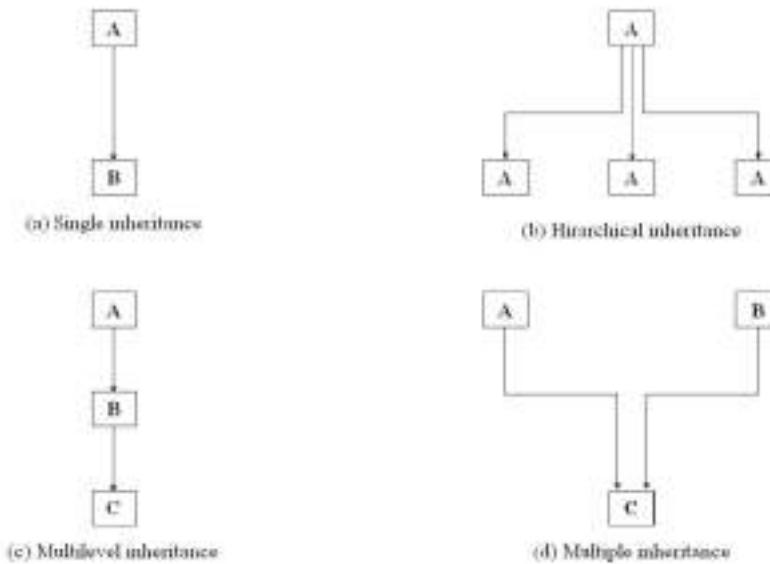


Fig. 6 Forms of Inheritance

Defining a subclass:

```
Class subclassname extends superclassname
{
    variables declaration;
    methods declaration;
}
```

The keyword `extends` indicates that the properties of the `superclassname` are extended to the `subclassname`.

VIII. Resources required(Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Write a program to implement single and multilevel inheritance.

Note: Attach the code at the end.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Justify: Java does not support multiple inheritance.
 2. Specify the conditions when the super keyword can be used.
 3. Write the importance of final variables and methods.
 4. Specify the conditions which needs to be satisfied while using the abstract classes.

(Space for answer)

XIII. Exercise:

- Exercise:**

 - 1 Develop a program to implement the multilevel inheritance.
 - 2 Develop a program to calculate the room area and volume to illustrate the concept of single inheritance (Assume suitable data wherever necessary).

(Space for answer)

.....
.....
.....

.....

.....

.....

.....

XIV. References/ Suggestions for Further Reading

1. <https://www.javatpoint.com/inheritance-in-java>
2. <https://www.geeksforgeeks.org/output-java-program-set-20-inheritance/>
3. <https://beginnersbook.com/2013/05/java-multiple-inheritance/>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related (35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 19: Develop a program for implementation of multiple inheritance.

I. Practical Significance:

Deriving a new class from multiple super/parent classes is known as multiple inheritance. Java doesn't allow multiple inheritance to avoid the ambiguity caused by the overriding methods of the classes. Interface is used to achieve multiple inheritance in java.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The Practical is expected to develop the following skills:

1. Write a program to define and extend the interfaces.
2. Develop a program to implement various forms of interfaces.

IV. Relevant Course Outcome(s)

Apply concept of inheritance for code reusability.

V. Practical Outcome (PrOs)

Develop a program for implementation of multiple inheritance.

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

1. Interfaces

Java does not support multiple inheritance.

Classes in java cannot have more than one superclass.

A large number of real-time applications require use of multiple inheritance where no. of methods are inherited from several classes. Multiple inheritance proves difficult and adds complexity to the language. Java provides an alternative approach called as **interfaces** to support to the concept of multiple inheritance.

Defining Interfaces

An interface is like a class. Interfaces also contain methods and variables but with a major difference. The difference is that interfaces define only **abstract methods and final fields**.

Interfaces do not specify any code to implement these methods and data fields contain only constants.

Syntax:

```
interface InterfaceName
{
    variables declaration;
    methods declaration;
}
```

Where interface is the keyword and **InterfaceName** is any valid java variable.

Variables are declared as:

static final type VariableName=Value;

Example: return_type methodName1(parameter_list);

Interface Definition:

Interface Item

```
{
    static final int id=100;
    static final String name="ABC";
    void display ();
}
```

Extending Interfaces

Interfaces can be extended. The new subinterface will inherit all the members of the superinterface.

interface name2 **extends** name1

```
{
    body of name2
}
```

Example:

interface ItemConstants

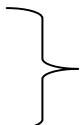
```
{
    int id=100;
    string name="ABC";
}
```



All constants in one interface

interface Item extends ItemConstants

```
{
    void display();
}
```



All methods in other interface

2. Multiple interfaces:

```

interface ItemConstants
{
    int id=100;
    string name="ABC";
}

interface ItemMethods
{
    void display( );
}

interface Item extends ItemConstants, ItemMethods
{
    .....
}

```

3. Implementing Interfaces:

Interfaces are used like superclasses whose features/properties are inherited by classes.

It is mandatory to define a class that inherits the given interface.

```

class classname implements interfacename
{
    body of classname
}

```

The class classname “implements” the interface interfacename.

class classname extends superclass implements interface1, interface2,....

```

{
    body of classname
}

```

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Demonstrate the use of interfaces to implement the concept of multiple inheritance.

Note: Attach the code at the end.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Differentiate between Class and Interface.
 2. Write similarities between interfaces and classes.
 3. Write advantages of interfaces.
 4. Will this code compile successfully?(Assume Suitable Class)

```
public interface FamousLine
{
    void ShowLine( )
    {
        System.out.println("Show Line");
    }
}
```

(Space for answer)

XIII. Exercise:

- 1 Correct the code to rectify the compile time error thrown.

```

interface NewShape
{
    void draw( );
}
interface Circle extends NewShape
{
    void getRadius( );
    int radius=10;
}
class NewCircle implements Circle
{
    public void getRadius( )
    {
        System.out.println(radius);
    }
}
class ExtendInterface extends NewCircle
{
    public static void main(String aregs[])
    {
        Circle nc=new NewCircle( );
        nc.getRadius( );
    }
}

```

- 2 Develop a program to find area of rectangle and circle using interfaces.
 3 Write output of the program.

```

interface Pet
{
    public void test();
}
class Dog implements Pet
{
    public void test()
    {
        System.out.println("Interface Method Implemented");
    }
    public static void main(String args[]){
        Pet p = new Dog();
        p.test();
    }
}

```

(Space for answer)

.....

.....

.....

.....

.....

.....

.....

XIV. References/ Suggestions for Further Reading

1. <https://www.geeksforgeeks.org/interfaces-in-java/>
2. Programming with java by E Balagurusamy

XV. Assessment Scheme

Performance Indicators		Weightage
Process related (35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related(35)	Product Related (15)	Total (50)	

Practical No. 20: Develop a program to import different classes in package.

I. Practical Significance:

Packages are collection of classes and interfaces. Importing of Packages which helps in Reusability. Better organization of the classes and interfaces helps in resolving the name conflicts.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The practical is expected to develop the following skills:

1. Write a program to create user defined packages.
2. Write a program to import the user defined packages.

IV. Relevant Course Outcome(s)- Develop program using OO method in java.

Apply concept of inheritance for code reusability.

V. Practical Outcome (PrOs)

Develop a program to import different classes in package.

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

What is Package

It is a collection of similar types of **classes, interfaces and sub-packages**.

Packages are categorized in two form, **built-in package** and **user-defined package**

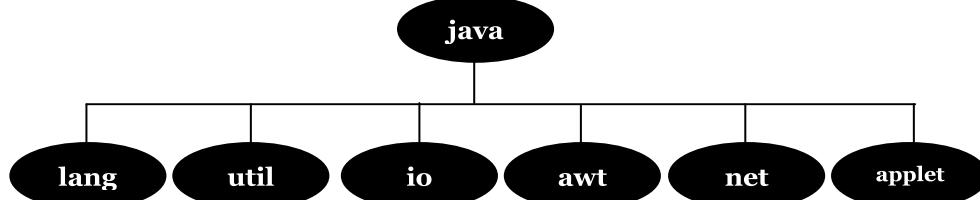


Fig. 7 Frequently Used API Packages

Using System Packages

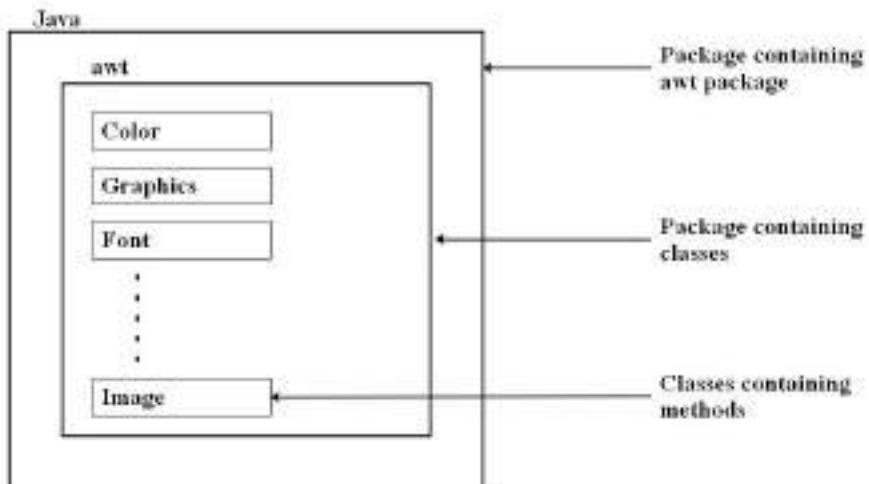


Fig. 8 Hierarchical representation of java.awt package

Accessing the classes stored in a package:

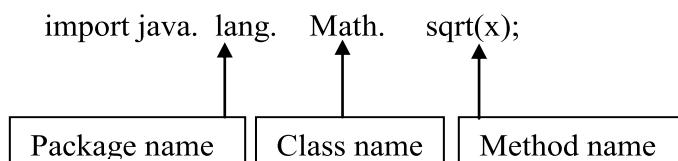
Method 1: Using import Statements:

1. The first method is to use the **Fully qualified class name** of the class:
`import pacakgenname.Classname;`

Example:

```
import java.lang.Math;
```

This statement imports the class `Math` and therefore class name can be used directly. It is not necessary to use the package name to qualify the class.



Method 2: Shortcut Approach

```
import pacakgenname.*;  
import java.lang.*;
```

This statement imports **every class** contained in the specified package.
Above statement will bring **all the classes** of `java.lang` package.

Creating Packages

```
package PackageName;           //package declaration  
public class FirstClass        // class definition  
{  
    -----// (body of class)  
}
```

Package Hierarchy

```
Package firstPackage.secondPackage;
```

Accessing a Package

A java system package can be accessed either using a **fully qualified class name** or using a **shortcut approach** through the import statement.

Syntax:

import package1 [. package2][.package3].classname;

package1 is the name of the **outer package**, package2 is the name of the package that is **inside the package1**.

Package hierarchy consists of any number of packages. Finally the explicit classname is specified.

Example:

import firstPackage.secondPackage.MyClass; // fully qualified class name

OR

import packagename.*;

import firstPackage.*; //shortcut approach

VIII. Resources required(Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Write a program to implement user defined packages in terms of creating a new package and importing the same.

Note: Attach the code at the end.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Name some of java packages.
 2. Can we import same class/package twice? Will the JVM load the package twice at run time?
 3. Write fully qualified and shortcut class naming approach with examples.
(Space for answer)

XIII. Exercise (Any two)

1. The code uses the class defined below. Class Importclass is not defined in circle folder. Will the code run without giving any errors?

```
import circle.NewCircle;
class ImportClass
{
    public static void main(String args[ ])
    {
        circle.NewCircle nc=new circle.NewCircle( );
        System.out.println("Hello Java");
    }
}
```

- Define a package named myInstitute include class named as department with one method to display the staff of that department. Develop a program to import this package in a java application and call the method defined in the package.
 - Develop a program which consists of the package named let_me_calculate with a class named calculator and a method named add to add two integer numbers. Import let_me_calculate package in another program (class named Demo) to add two numbers.

(Space for answer)

.....

.....

.....

.....

.....

.....

XIV. References/ Suggestions for Further Reading

1. <https://beginnersbook.com/2013/03/packages-in-java/>
2. <https://www.javatpoint.com/package>
3. Programming with java by E Balagurusamy

XV. Assessment Scheme

Performance Indicators		Weightage
Process related (35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 21 & 22: Develop a program for implementation of multithreading operation Part-I and Part-II.

I. Practical Significance:

Multithreading technique in java helps to run multiple programs or processes concurrently by utilizing the maximum CPU time. Multithreaded technique is implemented by creating, declaring, extending, implementing by thread. Student will be able to implement different types of thread methods by assigning the priority to illustrate simultaneous execution of thread operation.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The practical is expected to develop the following skills:

1. Develop program that enables programmers to do multiple things at one time using multiple threads.
2. Write programs to extend the thread and implement various thread methods.

IV. Relevant Course Outcome(s)

Develop programs using multithreading.

V. Practical Outcome (PrOs)

Develop a program for implementation of multithreading operation.

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Multithreading: Multithreading is a small program (process) which is divided into two or more subprogram (processes), that can be implemented simultaneously.

Thread: It is a small process which is used to divide a program into number of sub parts and each part can be executed in parallel.

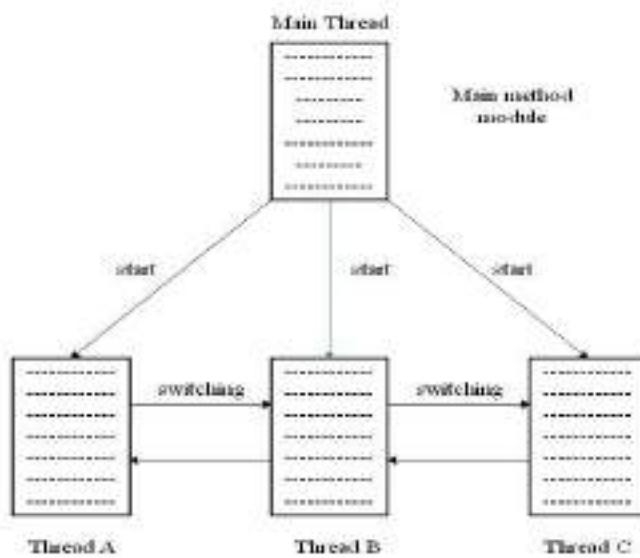


Fig. 10 A Multithreaded Program

Creating Threads

Threads are implemented using a method called `run()`.

Syntax:

```
public void run ()
{
    .....
}
```

Statements for
implementing
thread

A thread can be created in two ways:

1. By creating a thread class:

Define a class that extends `Thread` class and override its `run()`
Extending the thread Class

```
class MyThread extends Thread
{
    .....
}
```

2. By converting a class to a thread:

Define a class that implements `Runnable` interface. The `Runnable` interface has only one method, `run()`

`class A implements Runnable`

```
{
    .....
}
```

Synchronization

It means only a single thread can execute a block of code at the same time.

Example:

The method that will read information from a file and the method that will update the same file may be declared as synchronized.

```
synchronized void update()
{
    .....
    .....// synchronized code
}
```

Java Thread Method:

Sr. No.	Type	Method	Description
1	void	start()	It is used to start the execution of the <u>thread</u> .
2	void	run()	It is used to perform action for a thread.
3	static void	sleep()	It sleeps a <u>thread</u> for specified amount of time.
4	static Thread	currentThread()	It returns a reference to the currently executing <u>thread</u> object.
5	void	join()	It waits for a <u>thread</u> to die.
6	int	getPriority()	It returns the priority of the <u>thread</u> .
7	void	setPriority()	It changes the priority of the <u>thread</u> .
8	string	getName()	It returns the name of the <u>thread</u> .
9	void	setName()	It changes the name of the <u>thread</u> .
10	long	getID()	It returns the id of the <u>thread</u> .
11	boolean	isAlive()	It tests if the <u>thread</u> is alive.
12	static void	yield()	It causes the currently executing <u>thread</u> object to temporarily pause, allow other <u>threads</u> to execute.
13	void	suspend()	It is used to suspend the <u>thread</u> .
14	void	resume()	It is used to resume the suspended <u>thread</u> .
15	void	stop()	It is used to stop the <u>thread</u> .
16	void	destroy()	It is used to destroy <u>thread</u> group , all of its subgrp.
17	void	notify()	It is used to give the notification for only one <u>thread</u> which is waiting for a particular object.
18	void	notifyAll()	It is used to give the notification to all waiting <u>threads</u> of a particular object.

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1	Computer System with broad specifications			
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Develop a simple real-life application program to illustrate the use of multithreads.

Note: Attach the code at the end.

XI. Result (Output of Code):

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Is it possible to start a thread twice?
 2. Can we call the run() method instead of start()?
 3. Differentiate between notify() and notifyAll()?
 4. Explain the use of keyword synchronized.

(Space for answer)

XIII. Exercise:

1. Implement multithreading to perform simultaneous processes.
 2. The code given below calls the run() method of two threads while setting their priority. Will this code compile successfully? If not, correct the code.

```
class t1 extends Thread
{
    public void run( )
    {
        System.out.println("This is Thread1 class");
    }
}
Class t2 extends Thread
{
    public void run( )
    {
        System.out.println("This is Thread2 class");
    }
}

public class ThreadP
{
    public static void main(String args[ ])
    {
        t1 t=new t1();
        t2 tt=new t2();
        t.setPriority(Thread.MIN_PRIORITY);
        tt.setPriority(Thread.MIN_PRIORITY);
        t1.run();
        t2.run();
    }
}
```

3. Create three threads and run these threads according to set priority.

(Space for answer)

.....

.....

.....

.....

.....

XIV. References/ Suggestions for Further Reading

1. <http://www.javaprogrammingforums.com/cafe/11575-multithreading-importance-java.html>
2. Programming with Java by E. Balagurusamy
3. <https://www.javatpoint.com/multithreading-in-java>

XV. Assessment Scheme

Performance Indicators		Weightage
Process related (35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 23, 24 & 25: Develop a program for implementation of try, catch and finally block.

I. Practical Significance:

Managing errors and Exception handling helps to detect exceptional conditions in a program and fix the exceptions as and when they occur. Student will be able to handle different types of exceptions using try, catch and finally blocks.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The practical is expected to develop the following skills:

1. Develop a program to generate and process the exception using try and catch block.
2. Write a program to execute the finally block.

IV. Relevant Course Outcome(s)

Implement Exception Handling.

V. Practical Outcome (PrOs)

Develop a program for implementation of try, catch and finally block.

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Errors:

Errors are the mistakes that can make a program go wrong.

Error may produce a wrong results or abruptly terminates the execution of the program or may cause the system to crash.

Detecting and managing the errors is very important during the program execution.

Types of Errors:

1. Compile-time errors
2. Run-time errors

Exception Handling Tasks:

1. Find the Problem (Hit the exception)
2. Inform that an error has occurred (Throw the exception)
3. Receive the error information (Catch the exception)
4. Take corrective actions (Handle the exception)

Common Java Exceptions

Sr. No.	Exception Type	Cause of Exception
1	ArithmaticException	Caused by math errors such as division by zero
2	ArrayIndexOutOfBoundsException	Caused by bad array indexes
3	ArrayStoreException	Caused when a program tries to store the wrong type of data in an array
4	FileNotFoundException	Caused by an attempt to access a nonexistent file
5	IOException	Caused by general I/O failures, such as inability to read from a file
6	NullPointerException	Caused by referencing a null object
7	NumberFormatException	Caused when a conversion between strings and number fails
8	OutOfMemoryException	Caused when there's not enough memory to allocate a new object
9	SecurityException	Caused when an applet tries to perform an action not allowed by the browser's security setting
10	StackOverflowException	Caused when the system runs out of stack space
11	StringIndexOutOfBoundsException	Caused when a program attempts to access a nonexistent character position in a string

Categories of Exceptions:

1. Checked Exceptions

Checked exceptions are explicitly handled in the code itself using try catch blocks.

These are extended from the `java.lang.Exception` class

2. Unchecked Exceptions

Unchecked exceptions are not necessarily handled in the program code, instead the JVM handles such exceptions.

These are extended from the `java.lang.RuntimeException` class.

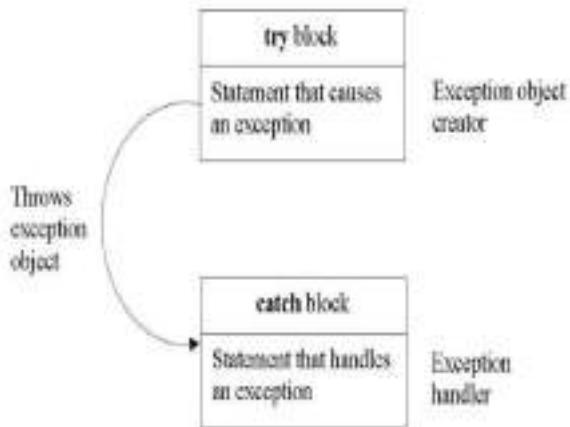
Exception Handling Code:

Try:

A keyword 'try' is used for a block of code that causes an error condition and 'throw' an exception

Catch:

A keyword 'catch' is used for a block of code that 'catches' the exception thrown by the 'try' block and handles it properly

**Fig. 12 Exception Handling Mechanism****Syntax:** Simple try, catch and finally statement

```

try
{
    Statement;          //generates an exception
}
catch(Exception-type e)
{
    Statement;          //processes the exception
}
.....
finally {.....} //optional

```

Finally Statement:-

finally statement handles an exception that is not caught by any of the earlier catch statements.

It is used to handle any exception generated within a try block.

It is written immediately after the try block or after the last catch block.

VIII. Resources required(Additional)**Nil****IX. Resources used (Additional)**

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Demonstrates exception handling using try, catch and finally block.

Note: Attach the code at the end.

(Space for answer)

XI. Result (Output of Code):

.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. How exception is thrown by main method?
 2. Differentiate between error and exception in java.
 3. Can we throw exception manually? Illustrate with sample program.
 4. Explain the use of finally block?

(Space for answer)

XIII. Exercise:

1. The program calculates sum of two numbers inputted as command-line arguments. When will it give an exception?

```
class excep
{
    public static void main(String args[])
    {
        try
        {
            int n=Integer.parseInt(args[0]);
            int n1=Integer.parseInt(args[1]);
            int n2=n+n1;
            System.out.println("Sum is:"+n2);
        }
        catch(NumberFormatException ex)
        {
            System.out.println(ex);
        }
        finally
        {
            System.out.println("You inputted a correct integer number");
        }
    }
}
```

2. Develop a program to accept a password from the user and throw “Authentication Failure” exception if the password is incorrect.
3. Write the exception thrown by the following code block?

```
Integer[][] ints = { { 1, 2, 3 }, { null }, { 7, 8, 9 } };
System.out.println("value = " + ints[1][1].intValue());
```

(Space for answer)

.....

XIV. References/ Suggestions for Further Reading

1. <https://www.javatpoint.com/finally-block-in-exception-handling>
2. https://www.tutorialspoint.com/java/java_exceptions.htm
3. Programming with Java by E. Balagurusamy

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 26 & 27: Develop a program for implementation of throw and throws clause.

I. Practical Significance:

The throw and throws clause is used to explicitly throw an exception from a method or any block of code. It is mainly used for throwing the custom exceptions (User defined exceptions). Students will be able to throw the user defined exceptions.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The practical is expected to develop the following skills:

1. Develop a program to throw our own exceptions i.e. user defined exceptions using throw.
2. Write a program to predict explicitly certain kinds of exception using throws clause.

IV. Relevant Course Outcome(s)

Implement Exception Handling.

V. Practical Outcome (PrOs)

Develop a program for implementation of throw & throws clause.

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Throwing Our Own Exceptions

throw: All methods in java use the ‘throw’ statement explicitly to **throw** an exception from a method or any block of code. The **throw** is a keyword in **java**. **throw** can be used for either checked or unchecked exception. It is mainly used to **throw** custom exceptions.

Syntax:

throw Throwable-instance;

where Throwable-instance must be an object of type Throwable or a subclass of Throwable.

throws: It is a keyword in java. Its is used to declare an exception. It is used with a method signature. Multiple exceptions can be declared through throws.

Syntax:

type method name(parameters) throws exception list

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Demonstrate use of throw and throws clause.

Note: Attach the code at the end.

XI. Result (Output of Code):

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Differentiate between throw and throws clause.
 2. In which situation the throws clause is used?
 3. Write the simple program for throwing our own exceptions.

(Space for answer)

XIII. Exercise:

1. Define an Exception called “NotMatchException” that is thrown when a string is not equal to “India”. Write a program that uses this exception.
 2. Write the output of the following program:

```
class tst
{
    public static void main(String args[])
    {
        Thread.sleep(10000);
        System.out.println("Hello java");
    }
}
```

3. Write the output of the following program:

```
class tst
{
    public static void main(String[] args) throws InterruptedException
    {
        Thread.sleep(10000);
        System.out.println("Hello Java");
    }
}
```

(Space for answer)

XIV. References / Suggestions for Further Reading

1. <https://www.geeksforgeeks.org/throw-throws-java/>
2. Programming with Java-A Primer by E Balagurusamy

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 28: Develop minimum two basic Applets. Display output with applet viewer and browser.

- a) **Develop a program on basic applet.**
- b) **Develop a program using control loops in applets**

I. Practical Significance:

Java applets help to make the web pages more dynamic. Java is also used to create small, dynamic programs that run along with or are embedded within Web pages. These programs (applets) can be used to display maps, weather, games or other interactive widgets or tools on a Web page.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline Knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and Practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The practical is expected to develop the following skills:

1. Write a program to build applet code.
2. Develop a program using different states of an applet and execute using applet viewer.

IV. Relevant Course Outcome(s)

Develop program using graphics and Applet.

V. Practical Outcome (PrOs)

Develop minimum two basic Applets. Display output with applet viewer and browser.

1. Develop a program on basic applet.
2. Develop a program using control loops in applets.

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Applets are small Java programs that can be used in Internet Computing. Applets can be transported over the Internet from one computer to another and run using the **Applet Viewer** or any Web browser that supports Java.

Through Applets Arithmetic operations, display graphics, play sounds, accept user input, animation and play interactive games can be performed.

java.awt.Component class

public void paint(Graphics g): It is used to paint the Applet. It contains Graphics class object which is used for drawing oval, rectangle, arc, etc.

Simple Program: The HelloJava applet

```
import java.awt.*;
import java.applet.*;
public class HelloJava extends Applet
{
    public void paint(Graphics g)
    {
        g.drawString("Hello Java",10,100);
    }
}
```

Output

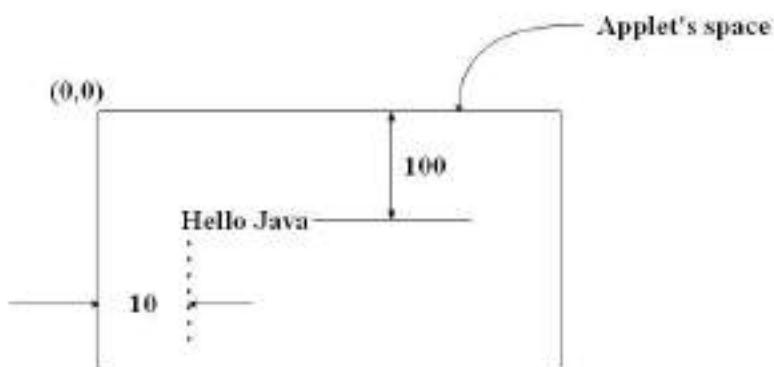


Fig. 14 Output of Program

Applet Tag

```
<APPLET
    CODE=HelloJava.class
    WIDTH=400
    HEIGHT=200>
</APPLET>
```

Adding Applet to HTML file

```
<HTML>
<! This page includes a welcome title in the title bar and also displays a welcome
message. Then it specifies the applet to be loaded and executed>
```

```

<HEAD>
    <TITLE>
        Welcome to Java Applets
    </TITLE>
</HEAD>
<BODY>
    <CENTER>
        <H1>Welcome to the World of Applets</H1>
    </CENTER>
    <BR>
    <CENTER>
        <APPLET
            CODE=HelloJava.class
            WIDTH=400
            HEIGHT=200>
        </APPLET>
    </CENTER>
</BODY>
</HTML>

```

Note: Save this file as **HelloJava.html** and save it in the same directory as the compiled applet.

Creating an Executable Applet:

Executable applet is obtained by compiling the source code of the applet i.e. **.class** file of the applet.

Compiling an applet is same as compiling an application. Java compiler is used to compile the applet.

HelloJava applet is stored in a file called **HelloJava.java**

Compile the program using **javac HelloJava.java**

The compiled output file is created called **HelloJava.class** (if no errors).

Running the Applet:

In above sample codes we have created **applet files** as well as the **HTML file containing the applet**, we must have the following files in our current directory.

HelloJava.java
 HelloJava.class
 HelloJava.html

There are two ways to run an applet:

1. By Java-enabled web browser tool(i.e. HTML file)
2. By Java appletviewer tool

1. By Java-enabled web browser tool (i.e. HTML file)

Through Java-enabled Web browser the entire web page can be seen containing the applet.

c:\>javac HelloJava.java
c:\>appletviewer HelloJava.java

2. By Java appletviewer tool

Through appletviewer tool, only applet output can be seen.

The appletviewer is not a full-fledged web browser and therefore it ignores all of the HTML tags except the part pertaining to the running of the applet.

To run the applet:

appletviewer HelloJava.html

	Applet Viewer : HelloJava.class
Applet	<p style="text-align: center;">Hello Java</p>
	appletloader.started

Fig.15. Output of HelloJava applet by using appletviewer

VIII. Resources required(Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Create two applets by using different ways in java.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Differentiate between init() and start()methods.
2. Explain use of start() and stop() method in applet life cycle?
3. The method paint() belongs to which class?

(Space for answer)

.....
.....
.....
.....
.....
.....
.....
.....

XIII. Exercise:

1. Develop a basic applet to display “Welcome to the World of Applet”.
 2. Develop a program to implement all methods of applet.
 3. Develop a program using control loops in applets.

(Space for answer)

Maharashtra State Board of Technical Education

.....

.....

.....

.....

.....

XIV. References/ Suggestions for Further Reading

1. https://www.tutorialspoint.com/javaexamples/applet_create.htm
2. <https://www.javatpoint.com/java-applet>
3. Programming with Java-A Primer by E Balagurusamy

XV. Assessment Scheme

Performance Indicators		Weightage
Process related(35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 29: Write a program to create animated shape using graphics and applets. You may use following shapes:

- a) Lines and Rectangles**
- b) Circles and Ellipses.**
- c) Arcs**
- d) Polygons with fill Polygon method.**

I. Practical Significance:

The Graphics class of java provides methods for drawing many different types of shapes. Students will be able to use different methods of graphics programming to draw simple lines, figures of different shapes, images and text in different fonts and styles with different appearances of colors.

II. Relevant Program Outcomes (POs)

- Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The practical is expected to develop the following skills:

1. Write programs by using different methods of graphics class.
2. Develop applet programs using graphics programming

IV. Relevant Course Outcome(s)

Develop program using graphics and Applet.

V. Practical Outcome (PrOs)

Write a program to create animated shape using graphics and applets. You may use following shapes:

1. Lines and Rectangles
2. Circles and Ellipses.
3. Arcs
4. Polygons with fill Polygon method.

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

Every applet has its own area of the screen known as canvas, where it creates its display.

The size of an applet's space is decided by the attributes of the <APPLET...> tag.

Drawing Methods of the Graphics Class:

Sr. No.	Method	Description	Syntax
1	drawArc()	Draws a hollow arc	void drawArc(int top, int left, int width, int height, int startAngle, int endAngle)
2	drawLine()	Draws a straight line	void drawLine(int startX, int startY, int endX, int endY)
3	drawOval()	Draws a hollow oval	void drawOval(int top, int left, int width, int height)
4	drawPolygon()	polygon	void drawPolygon(int x[], int y[], int numPointer)
5	drawRect()	Draws a rectangle	void drawRect(int top, int left, int width, int height)
6	drawRoundRect()	Draws a hollow rectangle with rounded corners	void drawRoundRect(int top, int left, int width, int height, int Xdiam, int Ydiam)
7	drawString()	Displays a text string	void drawString(String str, int x, int y)
8	fillArc()	Draws a filled arc	void fillArc(int top, int left, int width, int height, int startAngle, int sweepAngle)
9	fillOval()	Draws a filled oval	void fillOval(int top, int left, int width, int height)
10	fillPolygon()	Draws a filled polygon	void fillPolygon(int x[], int y[], int numPointer)
11	fillRect()	Draws a filled rectangle	void fillRect(int top, int left, int width, int height)
12	fillRoundRect()	Draws a filled rectangle with rounded corners	void fillRoundRect(int top, int left, int width, int height, int Xdiam, int Ydiam)
13	getColor()	Retrieves the current drawing color	void getColor(String nm)
14	getFont()	Retrieves the currently used font	void getFont()
15	getFontMetrics()	Retrieves the information about the font	void getFontMetrics()
16	setColor()	Sets drawing color	void setColor(Color c)
17	setFont()	Sets the font	void setFont()

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1				
2				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Write a program to implement an applet to draw basic animated shapes.

Note: Attach the code at the end.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Differentiate between executing the applet with appletviewer and HTML file.
 2. Explain methods required to draw different shapes with different colors?
 3. Differentiate between setforeground() and setColor() methods.
 4. Differentiate between applets and applications.

(Space for answer)

XIII. Exercise (Any two)

1. Develop a program to draw a polygon.
2. Develop an applet for drawing a human face.
3. Write the output of the program.

```
import java.awt.*;
import java.applet.*;
/*
<applet code="Arcs" width=300 Height=300>
</applet>
*/
public class Arcs extends Applet
{
    public void paint(Graphics g)
    {
        g.drawArc(10,40,70,70,0,75);
        g.fillArc(100,40,70,70,0,75);
        g.drawArc(10,100,70,80,0,175);
        g.fillArc(100,100,70,90,0,270);
        g.drawArc(200,80,80,80,0,180);
    }
}
```

(Space for answer)

XIV. References/ Suggestions for Further Reading

- References/ Suggestions for Further Reading**

 1. <https://www.javatpoint.com/>
 2. <https://javaproglang.blogspot.com/2014/03/java-graphics-programming.html#.W0sRDDlzbIU>
 3. Programming with Java-A Primer by E Balagurusamy

XV. Assessment Scheme

Performance Indicators		Weightage
Process related (35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 30: Develop a program to draw following shapes, graphics and applets.

- a) Cone**
- b) Cylinders**
- c) Cube**
- d) Square Inside a circle**
- e) Circle inside a square**

I. Practical Significance:

The Graphics class of java provides methods for drawing many different types of shapes. Students will be able to use different methods of graphics programming to draw cube, cylinders, cones, figures of different shapes, images and text in different fonts and styles with different appearances of colors.

II. Relevant Program Outcomes (POs)

- Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills

“Develop Applications using Java.”

The practical is expected to develop the following skills:

1. Write programs by using different methods of graphics class.
2. Develop applet programs using graphics programming.

IV. Relevant Course Outcome(s)

Develop program using graphics and Applet.

V. Practical Outcome (PrOs)

Develop a program to draw following shapes, graphics and applets.

1. Cone
2. Cylinders
3. Cube
4. Square Inside a circle
5. Circle inside a square

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

A java applet draws graphical image inside the space using the coordinate system. To draw different shapes, appropriate graphics method with proper syntax is required.

VIII. Resources required (Additional)

Nil

IX. Resources used (Additional)

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Demonstrate the use of basic applet, which includes different shapes like cone, cube, cylinders etc.

Note: Attach the code at the end.

XI. Result (Output of Code):

.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Which of these methods is a part of Abstract Window Toolkit (AWT)?
a) display() b) paint() c) drawstring() d) none of the above
Ans. _____

2. Enlist the methods required to draw cone/cylinder.

3. Explain the method with syntax to draw circle.

4. Differentiate Applet and application.

(Space for answer)

XIII. Exercise:

1. Write output of the program.

```
/*
<applet code="SetBackgroundColorExample" width=200 height=200>
</applet>
*/
import java.applet.Applet;
import java.awt.Color;
import java.awt.Graphics;
public class SetBackgroundColorExample extends Applet
{
    public void paint(Graphics g){
        /*
         * Set background color of an applet using
         * void setBackground(Color c) method.
         */
        setBackground(Color.red);
    }
}
```

2. Develop a program to draw any 2 of the following shapes:
a. Cone

- b. Cylinder
c. Cube

3. Develop a program to draw any one of the following:

 - a. Square Inside a circle
 - b. Circle inside a square

(Space for answer)

XIV. References/ Suggestions for Further Reading

1. <https://www.javatpoint.com/>
2. <https://javaproglang.blogspot.com/2014/03/java-graphics-programming.html#.W0sRDDlzbIU>
3. <https://www.java-examples.com/set-status-message-applet-window-example>
4. Programming with Java-A Primer by E Balagurusamy

XV. Assessment Scheme

Performance Indicators		Weightage
Process related (35 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.
4.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

Practical No. 31 & 32: Develop a program for implementation of I/O stream and file stream classes.

I. Practical Significance:

File handling is used to manage and store data which performs various operations. Reading and writing can be done at the level of bytes or characters.

Java supports byte stream classes that provide support for handling I/O operations on bytes and character stream classes for operations on characters.

II. Relevant Program Outcomes (POs)

- **Basic knowledge:** Apply knowledge of basic mathematics, sciences and basic engineering to solve the computer group related problems.
- **Discipline knowledge:** Apply Computer Programming knowledge to solve the computer group related problems.
- **Experiments and practice:** Plan to perform experiments and practices to use the results to solve the computer group related problems.
- **Engineering tools:** Apply relevant Computer programming technologies and tools with an understanding of the limitations.
- **Individual and team work:** Function effectively as a leader and team member in diverse/multidisciplinary teams.
- **Communication:** Communicate effectively in oral and written form.

III. Competency and Practical skills:

“Develop Applications using Java.”

This practical is expected to develop the following skills:

1. Write the program using file class to perform different file operations.
2. Develop the program to handle Input/Output Exceptions.

IV. Relevant Course Outcome(s)

Develop programs for handling I/O and file streams.

V. Practical Outcome (PrOs)

1. Develop a program for implementation of I/O stream classes.
2. Develop a program for implementation of file stream classes.

VI. Relevant Affective domain related Outcome(s)

1. Follow Safety Practices
2. Practice good housekeeping
3. Demonstrate Working as a leader/a team member.
4. Follow ethical practices.

VII. Minimum Theoretical Background

File:

A **file** is a collection of records placed in a particular area on the disk. A **record** is composed of several fields and field is a group of characters. Storing and managing data using files is known as **file processing** which includes tasks such as **creating files, updating files and manipulation of data**.

Streams:

In file processing, **input** refers to the flow into a program and **output** means the flow of data out of a program.

Input to a program may come from the **keyboard**, the **mouse**, the **memory**, the **disk**, a **network** or **another program**.

Output from a program may go to the **screen**, the **printer**, the **memory**, the **disk**, a **network**, or **another program**.

A stream in java is a path along which data flows (like a river or a pipe along which water flows). It has a source (of data) and a destination (for that data).

In Java, I/O operations are performed through streams.

Java streams are classified into two basic types:

1. Input Stream
2. Output Stream

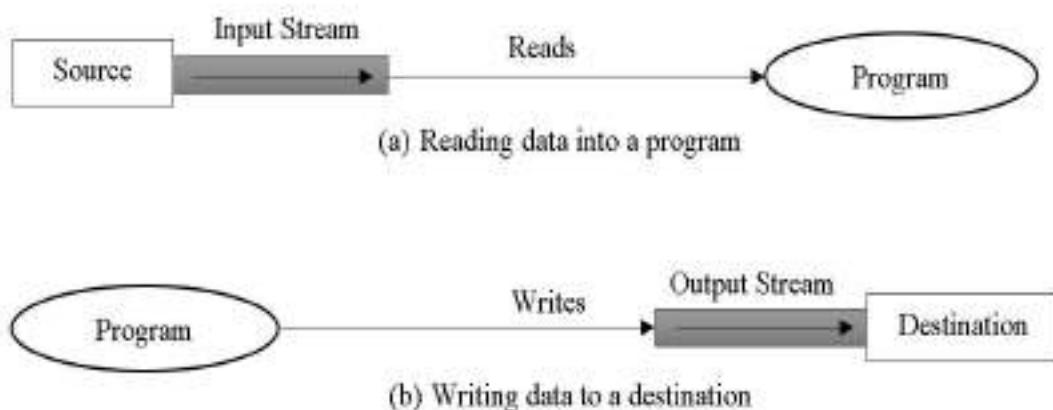


Fig. 16. Using Input and Output Streams

Categories of Stream Classes:

1. Byte stream classes
2. Character stream classes

VIII. Resources required

Nil

IX. Resources used

Sr. No.	Name of Resource	Broad Specification	Qty	Remarks (If any)
1				

X. Program Code: Teacher must assign a separate program statement to group of 3-4 students.

Demonstrate the use of stream classes for reading and writing bytes/characters.

Note: Attach the code at the end.

XI. Result (Output of Code):

.....
.....
.....
.....

XII. Practical Related Questions

Note: Below given are few sample questions for reference. Teacher must design more such questions so as to ensure the achievement of identified CO.

1. Enlist java stream classes,file operations.
2. Explain InputStream and OutputStream classes along with methods.
3. Explain Reader and Writer stream classes.
4. Find the compile-time error in the program given below.

```
import java.io.*;
class FileOutputStream
{
    public static void main (String args[])
    {
        FileOutputStream out;
        PrintStream p;
        try
        {
            Out=new FileOutputStream( );
            p=new PrintStream(out);
            p.println("This is written to a file");
            p.close( );
        }
        catch(Exception e)
        {
            System.err.println("Error writing to a file");
        }
    }
}
```

(Space for answer)

.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

XIII. Exercise:

1. Develop a program to copy characters from one file to another.
 2. Develop a program to write bytes to a file.
 3. Develop a program to display the content of file supplied as command line arguments.

(Space for answer)

XIV. References/ Suggestions for Further Reading

1. <https://stackoverflow.com>
2. <https://www.tutorialspoint.com>
3. Programming with Java Primer by E Balagurusamy

XV. Assessment Scheme

Performance Indicators		Weightage
Process related (10 Marks)		70%
1.	Logic formation	30%
2.	Debugging ability	30%
3.	Follow ethical practices	10%
Product related (15 Marks)		30%
4.	Expected output	10%
5.	Timely Submission	10%
6.	Answer to sample questions	10%
Total (50 Marks)		100%

List of Students /Team Members

1.
2.
3.

Marks Obtained			Dated signature of Teacher
Process Related (35)	Product Related (15)	Total (50)	

List Of Laboratory Manuals Developed by MSBTE

First Semester:

1 Fundamentals of ICT	22001
2 English	22101
3 English Work Book	22101W
4 Basic Science (Chemistry)	22102
5 Basic Science (Physics)	22102

Second Semester:

1 Bussiness Communication Using Computers	22009
2 Computer Peripherals & Hardware Maintenance	22013
3 Web Page Design with HTML	22014
4 Applied Science (Chemistry)	22202
5 Applied Science (Physics)	22202
6 Applied Machines	22203
7 Basic Surveying	22205
8 Applied Science (Chemistry)	22211
9 Applied Science (Physics)	22211
10 Fundamental of Electrical Engineering	22212
11 Elements of Electronics Engineering	22213
12 Elements of Electrical Engineering	22215
13 Basic Electronics	22216
14 C Language programming	22218
15 Basic Electronics	22225
16 Programming in C	22226
17 Fundamental of Chemical Engineering	22231

Third Semester:

1 Applied Multimedia Techniques	22024
2 Advanced Surveying	22301
3 Highway Engineering	22302
4 Mechanics of Structures	22303
5 Building Construction	22304
6 Concrete Technology	22305
7 Strength Of Materials	22306
8 Automobile Engines	22308
9 Automobile Transmission System	22309
10 Mechanical Operations	22313
11 Technology Of Inorganic Chemicals	22314
12 Object Oriented Programming Using C++	22316
13 Data Structure Using 'C'	22317
14 Computer Graphics	22318
15 Database Management System	22319
16 Digital Techniques	22320
17 Principles Of Database	22321
18 Digital Techniques & Microprocessor	22323
19 Electrical Circuits	22324
20 Electrical & Electronic Measurment	22325
21 Fundamental Of Power Electronics	22326
22 Electrical Materials & Wiring Practice	22328
23 Applied Electronics	22329
24 Electrical Circuits & Networks	22330
25 Electronic Measurements & Instrumentation	22333
26 Principles Of Electronics Communication	22334
27 Thermal Engineering	22337
28 Engineering Matrology	22342
29 Mechanical Engineering Materials	22343
30 Theory Of Machines	22344

Fourth Semester:

1 Hydraulics	22401
2 Geo Technical Engineering	22404
3 Chemical Process Instrumentation & Control	22407
4 Fluid Flow Operation	22409
5 Technology Of Organic Chemical	22410
6 Java Programming	22412
7 GUI Application Development Using VB.net	22034
8 Microprocessor	22415
9 Database Managment	22416
10 Electric Motors And Transformers	22418
11 Industrial Measurement	22420
12 Digital Electronic And Microcontroller Application	22421
13 Linear Integrated Circuits	22423
14 Microcontroller & Applications	22426
15 Basic Power Electronics	22427
16 Digital Communication Systems	22428
17 Mechanical Engineering Measurments	22443
18 Fluid Mechanics and Machinery	22445

19 Fundamentals Of Mechatronics	22048
20 Micro Project & Industrial Training Assessment Manual	22049

Fifth Semester:

1 Network Management & Administration	17061
2 Solid Modeling	17063
3 CNC Machines	17064
4 Behavioral Science(Hand Book)	17075
5 Behavioral Science (Assignment Book)	17075
6 Windows Programming using VC++	17076
7 Estimation and Costing	17501
8 Public Health Engineering	17503
9 Concrete Technology	17504
10 Design of Steel Structures	17505
11 Switchgear and Protection	17508
12 Microprocessor & Application	17509
13 A.C. Machines	17511
14 Operating System	17512
15 Java Programming	17515
16 System Programming	17517
17 Communication Technology	17519
18 Hydraulic & Pneumatics	17522
19 Advanced Automobile Engines	17523
20 Basic Electrical & Electronics	17524
21 Measurement and Control	17528
22 Power Engineering	17529
23 Metrology & Quality Control	17530
24 Computer Hardware & Networking	17533
25 Microcontroller	17534
26 Digital Communication	17535
27 Control System & PLC	17536
28 Audio Video Engineering	17537
29 Control System	17538
30 Industrial Electronics and applications	17541
31 Heat Transfer Operations	17560
32 Chemical Process Instrumentation & control	17561

Sixth Semester:

1 Solid Modeling	17063
2 Highway Engineering	17602
3 Contracts & Accounts	17603
4 Design of R.C.C. Structures	17604
5 Industrial Fluid Power	17608
6 Design of Machine Elements	17610
7 Automotive Electrical and Electronic Systems	17617
8 Vehicle Systems Maintenance	17618
9 Software Testing	17624
10 Advanced Java Programming	17625
11 Mobile Computing	17632
12 System Programing	17634
13 Testing & Maintenance of Electrical Equipments	17637
14 Power Electronics	17638
15 Illumination Engineering	17639
16 Power System Operation & Control	17643
17 Environmental Technology	17646
18 Mass Transfer Operation	17648
19 Advanced Communication System	17656
20 Mobile Communication	17657
21 Embedded System	17658
22 Process Control System	17663
23 Industrial Automation	17664
24 Industrial Drives	17667
25 Video Engineering	17668
26 Optical Fiber & Mobile Communication	17669
27 Therapeutic Equipment	17671
28 Intensive Care Equipment	17672
29 Medical Imaging Equipment	17673

Pharmacy Lab Manual

First Year:

1 Pharmaceutics - I	0805
2 Pharmaceutical Chemistry - I	0806
3 Pharmacognosy	0807
4 Biochemistry and Clinical Pathology	0808
5 Human Anatomy and Physiology	0809

Second Year:

1 Pharmaceutics - II	0811
2 Pharmaceutical Chemistry - II	0812
3 Pharmacology & Toxicology	0813
4 Hospital and Clinical Pharmacy	0816

HEAD OFFICE



Secretary,

Maharashtra State Board of Technical Education
49, Kherwadi, Bandra (East), Mumbai - 400 051

Maharashtra (INDIA)

Tel: (022)26471255 (5 -lines)

Fax: 022 - 26473980

Email: -secretary@msbte.com

Web -www.msbte.org.in

REGIONAL OFFICES:

MUMBAI

Deputy Secretary (T),
Mumbai Sub-region,
2nd Floor, Govt. Polytechnic Building,
49, Kherwadi, Bandra (East)
Mumbai - 400 051
Phone: 022-26473253 / 54
Fax: 022-26478795
Email: rbtmumbai@msbte.com

PUNE

Deputy Secretary (T),
M.S. Board of Technical Education,
Regional Office,
412-E, Bahirat Patil Chowk,
Shivaji Nagar, Pune
Phone: 020-25656994 / 25660319
Fax: 020-25656994
Email: rbtep@msbte.com

NAGPUR

Deputy Secretary (T),
M.S. Board of Technical Education
Regional Office,
Mangalwari Bazar, Sadar, Nagpur - 440 001
Phone: 0712-2564836 / 2562223
Fax: 0712-2560350
Email: rbteng@msbte.com

AURANGABAD

Deputy Secretary (T),
M.S. Board of Technical Education,
Regional Office,
Osmanpura, Aurangabad -431 001.
Phone: 0240-2334025 / 2331273
Fax: 0240-2349669
Email: rbteau@msbte.com