```
program no 1:
// we can not create object of Abstract classes
abstract class Parent
{
    abstract void m1();
    abstract void m2();
    void m3()
    {
        System.out.println("M3 method ");
    }
}
class Child extends Parent
{
    void m1()
    {
        System.out.println("M1 method ");
    }
    void m2()
    {
```

```
20          System.out.println("M2 method ");
21      }
22
23      public static void main(String[] args)
24      {
25          Parent p = new Parent ();
26
27      }
28  }
29  D:\AY 2023-24\SEM-I\JAVA\DIVB>javac AbstractDemo.java
30  AbstractDemo.java:23: error: Parent is abstract; cannot be ins
31  ============================================
32  program no 2:
33  // creating oject of child class to access and override abstra
34  abstract class Parent
35  {
36      abstract void m1();
37      abstract void m2();
38      void m3()
```

```
39      {
40          System.out.println("M3 method ");
41      }
42  }
43  class Child extends Parent
44  {
45      void m1()
46      {
47          System.out.println("M1 method ");
48      }
49      void m2()
50      {
51          System.out.println("M2 method ");
52      }
53
54      public static void main(String[] args)
55      {
56          Child c = new Child();
57          c.m1();
```

```
58          c.m2();
59          c.m3();
60
61      }
62  }
63  D:\AY 2023-24\SEM-I\JAVA\DIVB>java Child
64  M1 method
65  M2 method
66  M3 method
67  ================================================================
68  program no 3:
69  // using Parent reference variabe we can access overriding met
70  abstract class Parent
71  {
72      abstract void m1();
73      abstract void m2();
74      void m3()
75      {
76          System.out.println("M3 method ");
```

```
77      }
78  }
79  class Child extends Parent
80  {
81      void m1()
82      {
83          System.out.println("M1 method ");
84      }
85      void m2()
86      {
87          System.out.println("M2 method ");
88      }
89
90      public static void main(String[] args)
91      {
92          Child c = new Child();
93          c.m1();
94          c.m2();
95          c.m3();
```

```
 96
 97          Parent p = new Child();
 98          p.m1();
 99          p.m2();
100          p.m3();
101
102      }
103 }
104 D:\AY 2023-24\SEM-I\JAVA\DIVB>java Child
105 M1 method
106 M2 method
107 M3 method
108 M1 method
109 M2 method
110 M3 method
111
112 ================================================================
113
114                     Parent p = new Parent ();
```

```
                                    ^
1 error
================================================================
program no 4::
// multiple abstract classes
abstract class Parent
{
    abstract void m1();
    abstract void m2();
    void m3()
    {
        System.out.println("M3 method ");
    }
}
abstract class Child extends Parent
{
    // abstract void m2();
    void m1()
    {
```

```
134         System.out.println("M1 method ");
135     }
136
137 }
138 class Child2 extends Child
139 {
140     void m2()
141     {
142         System.out.println("M2 method ");
143     }
144     public static void main(String [] args)
145     {
146         Child2 c = new Child2();
147         c.m1();
148         c.m2();
149         c.m3();
150     }
151 }
152 D:\AY 2023-24\SEM-I\JAVA\DIVB>java Child2
```

```
153 M1 method
154 M2 method
155 M3 method
156 ============================
157 program no 5:
158 // multiple abstract classes
159 abstract class Parent
160 {
161     abstract void m1();
162     abstract void m2();
163     void m3()
164     {
165         System.out.println("M3 method ");
166     }
167 }
168 abstract class Child extends Parent
169 {
170     // abstract void m2();
171     void m1()
```

```
172     {
173         System.out.println("M1 method ");
174     }
175 
176 }
177 abstract class Child2 extends Child
178 {
179     // abstract void m2();
180 }
181 class Child3 extends Child2
182 {
183     void m2()
184     {
185         System.out.println("M2 method ");
186     }
187     public static void main(String [] args)
188     {
189         Child3 c = new Child3();
190         c.m1();
```

```
191             c.m2();
192             c.m3();
193         }
194 }
195
196 D:\AY 2023-24\SEM-I\JAVA\DIVB>java Child3
197 M1 method
198 M2 method
199 M3 method
200 ================================================
201 program no 6
202 // abstract class can have main()
203 abstract class Parent
204 {
205     public static void main(String [] args)
206     {
207         System.out.println("Main method ");
208     }
209 }
```

```
D:\AY 2023-24\SEM-I\JAVA\DIVB>java Parent
Main method
============================
program no 7:
// abstract class can have constructors

abstract class Parent
{

    Parent ()
    {
        System.out.println("abstract class parent constructor
    }
    abstract void m1();

}
class Child extends Parent
{
    Child()
```

```
229     {
230         super ();
231         System.out.println("Normal class Child  constructor ");
232     }
233     void m1()
234     {
235         System.out.println("Method m1");
236     }
237     public static void main(String [] args)
238     {
239         Child c = new Child();
240         c.m1();
241     }
242
243 }
244 D:\AY 2023-24\SEM-I\JAVA\DIVB>java Child
245 abstract class parent constructor
246 Normal class Child  constructor
247 Method m1
```

248
249
250
251
252
253
254
255