Q>
Syntax:

```
while(boolean){
      stmt-1;
      stmt-2
}
public class ExampleWhile{
      public static void main(String args[]){
            while(1){//1 is not boolean in java
                  System.out.println("hello");
            }
      }
}
```

A. CompileTime Error
B. Infinite times hello
C. hello
D. some problem by jvm during execution

Answer: A

Q>

```
public class ExampleWhile{
      public static void main(String args[]){
            while(true){
                  System.out.println("hello");
            }
      }
}
```

A. CompileTime Error
B. Infinite times hello
C. hello
D. some problem by jvm during execution

Answer: B

Q>
Syntax

```
while(true)
      stmt-1;
```

; -> it is a valid java statement

```
public class ExampleWhile{
      public static void main(String args[]){
            while(true);
      }
}
```

A. CompileTime Error
B. Infinite times hello
C. hello
D. some problem by jvm during execution
E. Infinite loop with no output

Answer: E

Q>

```
public class ExampleWhile{
      public static void main(String args[]){
            while(true)
                  int x  =10;//declarative statements are not allowed
      }
}
```

A. CompileTime Error

B. some problem by jvm during execution

C. Memory for x will be given 4bytes becoz of type int

D. None of the above

Anser: A

Q>

```
public class ExampleWhile{
      public static void main(String args[]){
            while(true){
                  int x  =10;
            }
      }
}
```

A. CompileTime Error

B. some problem by jvm during execution

C. Memory for x will be given 4bytes becoz of type int during execution

D. None of the above

Answer: C

Q>

```
public class ExampleWhile{
      public static void main(String args[]){
            while(true){
                  System.out.println("hello");//line-n1
            }
            System.out.println("hiee");//line-n2
      }
}
```

A. CompileTime Error at line-n1

B. hello infinite times

C. hiee

D. some problem by jvm during the execution

E. CompileTime Error at line-n2

F. None of the above

Answer: E(unreachable code)

Q>

```
public class ExampleWhile{
      public static void main(String args[]){
            while(false){
                  System.out.println("hello");//line-n1
            }
            System.out.println("hiee");//line-n2
      }
}
```

A. CompileTime Error at line-n1

B. hello

C. hiee

D. some problem by jvm during the execution

E. CompileTime Error at line-n2

F. None of the above

Answer: A(unreachable code)

Q>

```
public class ExampleWhile{
      public static void main(String args[]){
            int a =10,b=20;
            while(a<b){  10<20 ======> while(true)
                  System.out.println("hello");//line-n1
            }
            System.out.println("hiee");//line-n2
      }
}
```

A. CompileTime Error at line-n1
B. hello
C. hiee
D. some problem by jvm during the execution
E. CompileTime Error at line-n2
F. None of the above

Answer: F

Q>
Note: Whenever variables are marked as final, compiler will get to know the value of those varaibles and it will use the values in the
         Expression to get the result.

```
public class ExampleWhile{
      public static void main(String args[]){
            final int a =10,b=20;//compiler knows the value of the variables a, b.
            while(a<b){// while(10<20) ----> while(true)
                  System.out.println("hello");//line-n1
            }
            System.out.println("hiee");//line-n2
      }
}
```

A. CompileTime Error at line-n1
B. hello
C. hiee
D. some problem by jvm during the execution
E. CompileTime Error at line-n2
F. None of the above

Answer: E

Q>

```
public class ExampleWhile{
      public static void main(String args[]){
            final int a =10;
            while(a<20){// Compiler ====== while(10<20) ---> while(true)
                  System.out.println("hello");//line-n1
            }
            System.out.println("hiee");//line-n2
      }
}
```

A. CompileTime Error at line-n1
B. hello
C. hiee
D. some problem by jvm during the execution
E. CompileTime Error at line-n2
F. None of the above

Answer: E

Q>

```
public class ExampleWhile{
      public static void main(String args[]){
            int a =10;
            while(a<20){//JVM======> while(10<20) ====> while(true)
                  System.out.println("hello");//line-n1
            }
            System.out.println("hiee");//line-n2
      }
}
```

A. CompileTime Error at line-n1
B. hello
C. hiee
D. some problem by jvm during the execution
E. CompileTime Error at line-n2
F. hello infinite times
G. None of the above

Answer: F

Q>

```
public class ExampleDoWhile{
      public static void main(String args[]){
            do
                  System.out.println("hello");//line-n1
            while(true);
      }
}
```

A. CompileTime Error at line-n1
B. hello infinite times
C. some problem by jvm during the execution
D. hello
E. None of the above

Answer: B

Q>

Note: If we are using do while then there should be atleast one statement as the body otherwise it would result in "CompiletimeError".

```
public class ExampleDoWhile{
      public static void main(String args[]){
            do
            while(true);
      }
}
```

A. Compile time error
B. some problem by jvm during the execution
C. No output
D. None of the above

Answer: A

Q>

```
public class ExampleDoWhile{
      public static void main(String args[]){
            do;
            while(true);
      }
}
```

A. Compile time error

B. some problem by jvm during the execution

C. No output

D. None of the above

Answer: C

Q>

```
public class ExampleDoWhile{
      public static void main(String args[]){
            do
                  int x =10; //line-n1
            while(true);
      }
}
```

A. Compile time error at line-n1

B. some problem by jvm during the execution

C. No output

D. None of the above

Answer: A

Q>

```
public class ExampleDoWhile{
      public static void main(String args[]){
            do{
                  int x =10; //line-n1
            }while(true);
      }
}
```

A. Compile time error at line-n1

B. some problem by jvm during the execution

C. No output

D. None of the above

Answer: C

Q>

```
public class ExampleDoWhile{
      public static void main(String args[]){
                  do while(true)
                  System.out.println("hello");
                  while(true);
      }
}
```

A. CompileTime Error

B. hello infinite times

C. some problem by jvm during the execution

D. hello

E. None of the above

Answer: B(refer to the changes made by compiler as shown below)

Compiler will do

```
do
      while(true)
                  System.out.println("hello");
while(true);
```