

3.1 関係演算子と条件

Java では場合に応じた処理を行うことができます。式の値が真 (true) と偽 (false) のいずれかであらわされるものを条件といいます。条件を作るために使う記号を関係演算子といいます。

表 3.1 関係演算子一覧

演算子	式が true
==	右辺が左辺に等しい
!=	右辺が左辺に等しくない
.	右辺より左辺が大きい
>=	右辺が左辺以上
<	右辺より左辺が小さい
<=	右辺が左辺以下

3.2 if 文

条件判断文のひとつに if 文があります。if 文は条件が true の場合に指定した文を処理する構文です。

Java では場合に応じた処理を行うことができます。式の値が真 (true) と偽 (false) のいずれかであらわされるものを条件といいます。条件を作るために使う記号を関係演算子といいます。

Sample011

```
import java.io.*;
public class Sample011{
public static void main(String[] args) throws IOException{
System.out.println("商を求めます。整数を 2 つ入力してください。");
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
String str1=br.readLine();
String str2=br.readLine();
int num1=Integer.parseInt(str1);
int num2=Integer.parseInt(str2);
```

```
if(num2==0)
System.out.println(“割る数として 0 が入力されました。処理を終了します。”);
else{
ans=num1/num2;
System.out.println(num1+”と”+num2+”が入力されました。商は”+ans+”となります。”);
}
}
}
}
```

商の計算を行う場合、割る数は 0 以外の数である必要があります。Sample011 では、割る数が 0 であった場合に計算を行うことなく処理を終了するプログラムを記述しています。

【line10-11】

if 文では条件が true であった場合に続けて記述された 1 文が実行されます。Sample011 では num2 の値が 0 であった場合に line11 が実行されることとなります。num2 の値が 0 以外の場合は line11 は実行されません。

【line12-15】

if 文では条件が false だった場合の処理を指定することができます。その場合は else を用い、else に続けて条件が false の場合に実行したい文を記述します。Sample011 では num2 の値が 0 以外の場合に else に続く {} 内の処理が実行されます。

なお、if、else とともにそれぞれの条件で実行したい文が複数ある場合は、その部分を {} でくくる必要があります。(次に説明する else if の場合も) となります。

if 文では複数の条件を判断させて処理することもできます。

Sample012

```
import java.io.*;
public class Sample012{
public static void main(String[] args) throws IOException{
System.out.println(“試験の点数を入力してください。”);
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
String str1=br.readLine();
int num1=Integer.parseInt(str1);
if(num1=>80)
System.out.println(“成績 : A”);
```

```
else if(num1=>70)
System.out.println(“成績 : B”);
else if(num1=>60)
System.out.println(“成績 : C”);
else
System.out.println(“成績 : D”);
}
}
```

【line8-13】

num1 が 80 以上の場合には、続く line9 の処理をおこない、以降の else if の条件判断や処理はおこないません。num1 が 80 より小さい場合は line9 の処理をおこなわずに line10 の条件の判断に移ります。

【line14】

それまでの if や else if のどの条件にもあてはまらなかった場合に、else に続く処理をおこないません。なお else を必ずしも記述する必要はありません。

3.3 switch 文

if 文の他に条件によって処理をおこなう構文として switch 文があります。

switch 文では switch 文内の式が case の値と一致すればその後の文から break までの処理をします。どの条件にも当てはまらない場合は default 以下の文を処理します。

Sample013

```
import java.io.*;
public class Sample013{
public static void main(String[] args) throws IOException{
System.out.println(“試験の点数を入力してください。”);
BufferedReader br=new BufferedReader(new InputStreamReader(System.in));
String str1=br.readLine();
int num1=Integer.parseInt(str1);
int rank/=10;
switch(rank){
case 10:
System.out.println(“成績 : A”);
break;
```

```
case 9:
System.out.println("成績 : A");
break;
case 8:
System.out.println("成績 : A");
break;
case 7:
System.out.println("成績 : B");
break;
case 6:
System.out.println("成績 : C");
break;
default:
System.out.println("成績 : D");
break;
}
}
}
```

Sample013 は Sample012 と同様の処理を if 文ではなく switch 文を用いて書き換えたものです。

【line8】

int 型の変数 num1 を 10 で割った値を新しい int 型の変数 rank に代入しています。

【line9】

switch 文では switch の記述の直後の()内の式が case の後の値と一致する場合に、その後の文から break までの文を処理します。なお switch 文では条件判断のために case の後に記述する値は整数か文字でなければなりません。

【line25】

いずれとも一致しない場合は default 以下の分を処理します。なお if 文における else と同様、default は必ずしも記述する必要はありません。

3.5 論理演算子

論理演算子を用いることでより複雑な条件を記述することができます。Java では主に&&演算子と||演算子を用います。

数学で考えると&&演算子は積集合 (and)、||演算子は和集合 (or) に相当します。

表 3.1 真理値表

左辺	右辺	&&	
false	false	false	true
false	true	false	true
true	false	false	true
true	true	true	false

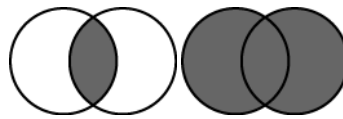


図 3.1 論理演算子&& (左) と|| (右)

3.7 演習

3.7.1 整数を入力させ、偶数か奇数かを判別するプログラムを記述しなさい。

- ・ (例) 2 を入力した場合に「2 が入力されました。2 は偶数です。」と表示される。
- ・ 0 が入力された場合に「0 が入力されました。再度整数を入力してください。」と表示され、数値入力に移る (再度 0 が入力された場合にはプログラム終了とする)。

3.7.2 試験の点数として二つの整数を入力させ (100~0)、成績を評価するプログラムを記述しなさい。

- ・ 100~0 以外の数が入力された場合に「誤った値が入力されました。再度値を入力してください。」と表示される。
- ・ 評価の基準は
 - A : それぞれが 60 点以上でかつ合計が 150 点以上
 - B : A の条件に該当せず合計が 120 点以上
 - C : B の条件に該当せずそれぞれが 20 点以上でかつ合計が 80 点以上
 - D : C の条件に該当せず

3.7.3 年月日を入力して曜日を表示させるプログラムを記述しなさい。

- ・ 西暦が 400 で割り切れる年はうるう年である。
- ・ 400 で割り切れない場合、西暦が 100 で割り切れる年はうるう年ではない。
- ・ 100 で割り切れない場合、西暦が 4 で割り切れる年はうるう年である。
- ・ 4 で割り切れない場合、うるう年ではない。