順序結構: 程序從上到下執行

分之結構:

If-else if-else

switch-case

循環結構:

for

while

do-while

1. if-else條件判斷結構

結構一:

if(條件表達式){

執行表達式

}

結構二: 二選一

if(條件表達式){

執行表達式1

}else{

執行表達式2

}

結構三: n選一

if(條件表達式){

執行表達式1

}else{

執行表達式2

}else{

執行表達式3

}…

else{

執行表達式n

}

* 1. 說明：

1. else 結構式是可選的。
2. 針對於條件表達式：

 如果多個條件表達式之間有＂互斥＂關係（或沒有交集的關係），哪個判斷何執行聲明在上面還是下面，無所謂。

 如果多個條件表達式之間有交集的關係,需要根據實際情況,考慮清楚應該將哪個結構聲明在上　  
　 面。

 如果多個條件表達式之間有包含的關係，通常情況下，需要將範圍小的聲明在範圍大的上面。否  
　 則，範圍小的就沒機會執行了。

1. if-else結構是可以相互嵌套的。
2. 如果if-else結構中的執行語句只有一行時，對應的一對｛｝可以省略的。但是，不建議大家省略。
3. switch-case選擇結構

 switch(表達式){

case 常量1:

執行語句1;

//break;

case 常量2:

執行語句2;

//break;

…

default:

執行語句N;

//break;

 說明：

1. 根據switch表達式中的值，依次匹配各個case中的常量。一旦匹配成功，則進入相應case結構中，調用其執行語句。  
   當調用完執行語句以後，則仍然繼續向下執行其它case結構中的執行語句，直到遇到break關鍵字或此switch-case結構a末尾結束為止。
2. break，可以使用在switch-case結構中，表示一旦執行到此關鍵字，就跳出switch-case結構。
3. switch結構中的表達式，只能是如下的六種數據類型之一：  
   byte、short、char、int、枚舉類型變量、String類型。
4. case 之後只能聲明常量。不能聲明範圍。
5. break關鍵字是可選的。
6. default：相當於if-else結構中的else。default結構是可選的，而且位置是靈活的。

 如果在switch-case結構中的多個case的執行語句相同，則可考慮合併。

　break在switch-case中是可選的。