

```
while (a > b) { a--; }
```

будет постоянно уменьшать a , проверяя перед каждой итерацией, что a больше b.

```
do { a--; } while (a > b);
```

будет постоянно уменьшать a, проверяя после каждой итерации, что a больше b.

```
for (int a = 0; a < 10; a++) { stdout.printf("%d\n", a); }
```

инициализирует a нулем, затем постоянно выводит значение a до тех пор, пока a меньше десяти, увеличивая a в каждой итерации.

```
foreach (int a in int_array) { stdout.printf("%d\n", a); }
```

будет выводить каждое число из массива или другой перечисляемой коллекции. Смысл слова "перечисляемой" будет описан позже.

Все четыре типа циклов могут управляться ключевыми словами break и continue. Выражение break вызовет немедленное завершение цикла, continue указывает, что надо перейти к следующей итерации.

```
if (a > 0) { stdout.printf("a is greater than 0\n"); }  
else if (a < 0) { stdout.printf("a is less than 0\n"); }  
else { stdout.printf("a is equal to 0\n"); }
```

выполнит определенную часть кода, основываясь на наборе условий. Первое условие, соответственно, решает, какой код будет выполнен; если a больше нуля, то проверка меньше ли a нуля производится не будет. Можно добавлять любое количество блоков else if и один или ноль блоков else.

```
switch (a) {  
    case 1:  
        stdout.printf("one\n");  
        break;  
    case 2:  
    case 3:  
        stdout.printf("two or three\n");  
        break;  
    default:
```

```
    stdout.printf("unknown\n");  
    break;  
}
```

Выражение switch выполняет точно одну или ноль секций кода, в зависимости от переданного значения. В Vala нет "проваливания" от одного case к другому, кроме случаев, когда после case нет кода. Чтобы обеспечить это, каждый непустой case должен заканчиваться выражением break, return или throw. Возможно использование выражений switch со строками.

Замечание для C-программистов: условия всегда должны возвращать логическое значение. Это значит, что если вы хотите проверить переменную на равенство null или нулю, вы должны сделать это явно: if (object != null) { } или if (number != 0) { }.