Pascal Cheatsheet

Pascal is a procedural programming language that was designed to encourage good programming practices and readability. It is widely used in education and scientific computing for tasks such as numerical analysis, data processing, and simulation.

Unique Features

- Strong typing
- Structured programming constructs
- Modular programming
- Pointers and dynamic memory allocation
- Object-oriented programming (in some dialects)

Variables

Variables in Pascal are declared using the `var` keyword. Pascal supports static typing, so you need to specify the type of the variable.

```pascal
var
  name: string;
  age: integer;
  pi: double;
```

Functions

Functions in Pascal are declared using the `function` keyword followed by the function name and parameters. Pascal supports nested functions, which are functions that are defined inside other functions.

```pascal
function greet(name: string): string;
begin
  greet := 'Hello, ' + name + '!';
end;

writeln(greet('John'));

var
  add: function(a, b: integer): integer;

add := function(a, b: integer): integer
begin
  add := a + b;
end;

writeln(add(2, 3));
```

Loops

Pascal supports `for`, `while`, and `repeat until` loops, as well as the `if`, `else if`, and `else` statements.

```
var
  numbers: array[1..5] of integer;
  i: integer;

numbers := (1, 2, 3, 4, 5);

for i := 1 to 5 do
  writeln(numbers[i]);

i := 0;
while i < 5 do
begin
  writeln(i);
  i := i + 1;
end;

i := 0;
repeat
  writeln(i);
  i := i + 1;
until i >= 5;
```

Conditionals

Pascal supports `if`, `else if`, and `else` statements, as well as the ternary operator (in some dialects).

```
var
  age: integer;
  result: string;

age := 30;

if age < 18 then
  writeln('You are too young to vote.')
else if age < 21 then
  writeln('You can vote, but not drink.')
else
  writeln('You can vote and drink.');

if age >= 18 then
  result := 'You are an adult'
else
  result := 'You are not an adult';

writeln(result);
```

File Manipulation

Pascal provides several ways to read and write files. You can use the `Assign` , `Rewrite` , `Reset` , `Write` , `Read` , and `Close` functions to create, read, write, and delete files.

```
var
  fileHandle: text;
  content: string;

Assign(fileHandle, 'example.txt');
Rewrite(fileHandle);
Write(fileHandle, 'Hello, world!');
Close(fileHandle);

Reset(fileHandle);
Read(fileHandle, content);
Close(fileHandle);
writeln(content);

Erase(fileHandle);
```

Resources

- Free Pascal Documentation
- Pascal Style Guide
- Pascal Tutorials
- Pascal Programming for Schools
- Pascal Compiler Online