### Declarations

<table>
<thead>
<tr>
<th>Dim, Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dim B, C As Byte</td>
<td>B is Variant, C is 0..255</td>
</tr>
<tr>
<td>Integer</td>
<td>16 bit, -32,768 to 32,767</td>
</tr>
<tr>
<td>Long</td>
<td>32 bit, -2,146,948,408 to 2,146,948,407</td>
</tr>
<tr>
<td>Currency</td>
<td>64 bit, -9,223,372,036,854,775,808 to 9,223,372,036,854,775,807</td>
</tr>
<tr>
<td>Single</td>
<td>32 bit, -3.4E38 to 3.4E38, 6 digits</td>
</tr>
<tr>
<td>Double</td>
<td>64 bit, -1.0E308 to 1.0E308, 15 digits</td>
</tr>
<tr>
<td>Date</td>
<td>Days since 1/1/1900</td>
</tr>
<tr>
<td>Object</td>
<td>Reference to any object</td>
</tr>
<tr>
<td>Variant</td>
<td>Any of the types or Null, Empty, Nothing, Special values</td>
</tr>
<tr>
<td>String</td>
<td>Variable length, max 2E9 characters</td>
</tr>
<tr>
<td>Integer 16 bit</td>
<td>-32,786 to 32,767</td>
</tr>
<tr>
<td>Date/time</td>
<td>#10/24/02# 24th Oct 2002</td>
</tr>
<tr>
<td>Time</td>
<td>#10/24/02 14:15:00# 24th Oct 02 at 14:15</td>
</tr>
</tbody>
</table>

### Procedures = Subroutines and Functions

<table>
<thead>
<tr>
<th>Subroutine/Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dim custTable(i) As Customer</td>
<td>Element in collection</td>
</tr>
<tr>
<td>custID As Long</td>
<td>Property=Control in module</td>
</tr>
<tr>
<td>custName As String</td>
<td>Property in subform</td>
</tr>
<tr>
<td>custAddress As String</td>
<td>Control in main form</td>
</tr>
<tr>
<td>custCommon.simDate</td>
<td>Variable in foreign module</td>
</tr>
<tr>
<td>custID, custTable(i)</td>
<td>Index array in field of records</td>
</tr>
<tr>
<td>Me.Recordset</td>
<td>Apply before dot and bang</td>
</tr>
<tr>
<td>.addr &amp; zip</td>
<td>= Null</td>
</tr>
<tr>
<td>.addr</td>
<td>Exit not allowed</td>
</tr>
<tr>
<td>Dim custTable(20) As Customer</td>
<td>Element in collection</td>
</tr>
<tr>
<td>custID As Long</td>
<td>Property=Control in module</td>
</tr>
<tr>
<td>custName As String</td>
<td>Property in subform</td>
</tr>
<tr>
<td>custAddress As String</td>
<td>Control in main form</td>
</tr>
<tr>
<td>custCommon.simDate</td>
<td>Variable in foreign module</td>
</tr>
<tr>
<td>custID, custTable(i)</td>
<td>Index array in field of records</td>
</tr>
</tbody>
</table>

### Operators, decreasing precedence

- `&` Concatenate strings
- `+` Add and subtract
- `/` Divide, Single or Double result
- `*` Multiply, Result type is Integer, Double, etc.
- `<` Equal, unequal, less than, etc.
- `<` Less than
- `=` Equal
- `<>` Not equal
- `>` Greater than
- `>=` Greater than or equal to
- `<=` Less than or equal to
- `>` Greater than
- `>=` Greater than or equal to
- `<=` Less than or equal to

### Addressing

<table>
<thead>
<tr>
<th>Addressing</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Me.Parent.txtName</td>
<td>Control in main form</td>
</tr>
<tr>
<td>custTable(i).custID</td>
<td>Field in array of records</td>
</tr>
<tr>
<td>With Me.Recordset</td>
<td>Apply before dot and bang</td>
</tr>
<tr>
<td>.addr &amp; zip</td>
<td>= Null</td>
</tr>
<tr>
<td>.addr</td>
<td>Exit not allowed</td>
</tr>
</tbody>
</table>

### Conditional statements

- `If a=2 Then` Case 4000
- `Case 4001, 5000 To 5999` Case 4001, 5000 To 5999
- `ElseIf a=2 Then` Case 4000
- `Else` Case 4001, 5000 To 5999
- `End Select` Case 4001, 5000 To 5999
- `End If` Case 4001, 5000 To 5999
- `End Sub` Case 4001, 5000 To 5999

### Loops

- `Do While` Loop while condition
- `Do While` Loop while condition
- `Do While` Loop while condition
- `Do While` Loop while condition
- `Do While` Loop while condition

### VBA Reference Card Errors

- **String functions**
  - Null parameters: A Null string as input will give the result Null. Null as another parameter is an error.
  - Asc(“ABC”) = 65, Ascii code for first character
  - Len(“A, B”) = 3, length of string.
  - Left(“abc”, 2) = “ab”, leftmost two characters
  - Left(“abc”, 8) = “abc”, as many as available
  - Right(“abc”, 2) = “bc”, rightmost two characters
  - Mid(“abcdef”, 2, 3) = “bcd”, three characters starting at second character
  - LTrim(“ “) = “”, leading spaces removed
  - RTrim(“ “) = “”, trailing spaces removed
  - Trim(“ “) = “”, both leading and trailing removed
  - LCase(“A-B”) = “a-b”, lower case of all letters
  - UCase(“A-B”) = “A-B”, upper case of all letters
  - Space(5) = String of 5 spaces

- **Option Compare**
  - Option in start of module. Text: string comparison is case insensitive, as is VBA. SQL: string comparison is case sensitive. SQL can be case insensitive.

- **IIf and Choose**
  - IIf(a=a, b, c) = b
  - IIf(Null, b, c) = c
  - IIf(a, b, c) = b
  - IIf(Null, b, c) = c
  - Choose(a, c) = b
  - Choose(4, a, c) = Null
  - Choose(Null, a, b, c) = Error

- **Array bounds**
  - LBound(d) = Lower bound for first index
  - LBound(d, 2) = Lower bound for second index
  - UBound(d) = Upper bound for first index
  - UBound(d, 3) = Upper bound for third index
**Type check functions**

- **IsArray(v)** Tests for any type of array
- **IsDate(v)** Tests whether v is a date value, or is a text that can be converted to a date
- **IsError(v)** Tests whether v is an error code
- **IsMissing(v)** Tests whether v is a parameter that is missing in the current call.
- **IsNull(v)** Tests whether v is unallocated (Strings of length 0 are not Null)
- **IsNumeric(v)** Tests whether v is a numeric type
- **IsObject(v)** Tests whether v is a reference to an object, for instance a Form. True also if v is Nothing (the nil-pointer)

**Math functions**

- **Abs(x)** Returns x for x >= 0, -x otherwise.
- **Int(x)** Rounds x down to nearest integral value
- **Fix(x)** Rounds x up to nearest integral value
- **Sqr(x)** Square root of x. Sqr(9) = 3.
- **Sin(x)**, **Cos(x)**, **Tan(x)**, **Atn(x)** Trigonometric functions.

**Financial functions**

- **NPV(0.12, d( ) )** The array d must be of type Double and contain a payment list of payments. Returns the net present value of these payments at an interest rate of 0.12, i.e. 12%.
- **IRR(d( ), 0.1)** The second parameter is a guess at the interest rate, to allow IRR to find a reasonable result.

**Other**

- **Step into F8 Next menu/tab Ctrl+Tab**
- **Step over Shift+F8 Next application Tab Ctrl+Tab**
- **Go to source code F9**
- **Close VBA/Appl Alt+F4**
- **Close Form Ctrl+F4**
- **Close APPB Editor F12**

**Control prefixes**

- **db** Database
- **chk** CHECKBOX
- **cmd** COMMAND Button
- **dl** OTHER
- **grp** GROUP Box
- **opt** OPTION Button

**Named formats**

- "Currency", "Short Date"