

Why Python?

There are many different programming languages available for the modern computer, and some still available for older 8 and 16-bit computers too. Some of these languages are designed for scientific work, others for mobile platforms and such. So why choose Python out of all the rest?

PYTHON POWER

Ever since the earliest home computers were available, enthusiasts, users and professionals have toiled away until the wee hours, slaving over an overheating heap of circuitry to create something akin to magic.

These pioneers of programming carved their way into a new frontier, forging small routines that enabled the letter 'A' to scroll across the screen. It may not sound terribly exciting to a generation that's used to ultra high-definition graphics and open world, multiplayer online gaming. However, forty-something years ago it was blindingly brilliant.

Naturally these bedroom coders helped form the foundations for every piece of digital technology we use today. Some went on to become chief developers for top software companies, whereas others pushed the available hardware to its limits and founded the billion pound gaming empire that continually amazes us.

Regardless of whether you use an Android device, iOS device, PC, Mac, Linux, Smart TV, games console, MP3 player, GPS device built-in to a car, set-top box or a thousand other connected and 'smart' appliances, behind them all is programming.

All those aforementioned digital devices need instructions to tell them what to do, and allow them to be interacted with. These instructions form the programming core of the device and that core can be built using a variety of programming languages.

The languages in use today differ depending on the situation, the platform, the device's use and how the device will interact with its

```
🔀 Bombs - GUI - ThelDE - [d:\uppsrc\CtrlLib\ArrayCtrl.cpp windows-1252] { examples }
File Edit Macro Project Build Debug Assist Setup
                         X 🖺 📵 GUI
                                                  SetCursor(p.y);
Bombs
                 plugin/bmp
                                          Ctrl::ChildGotFocus();
CtrlLib
                 plugin\z
                                                                                                                                                    (II)
CtrlCore
                 plugin\png
RichText
                 cprj-aux>
                                     void ArrayCtrl::ChildLostFocus()
PdfDraw
                 <ide-aux>
Draw Draw
                 = <temp-aux>
                                          if(cursor >= 0)
Core Core
                                              RefreshRow(cursor);
                                          Ctrl::ChildLostFocus();
# EditCtrl.h
                C AKeys.cpp
C EditField.cpp
                # RichText.h
                                     void ArrayCtrl::Paint(Draw& w) {
# TextEdit.h
                RichTextView.cpp
                                          LTIMING("Paint");
C Text.cpp
                C Prompt.cpp
                                          Size size = GetSize();
                C Help.cpp
C LineEdit.cpp
                                          Rect r;
C DocEdit.cpp
                # DateTimeCtrl.h
                                          r.bottom = 0;
# ScrollBar.h
                C DateTimeCtrl.cpp
                                          bool hasfocus = HasFocusDeep();
 ScrollBar.cpp
                Bar
                                          int i = GetLineAt(sb);
                # Bar.h
# HeaderCtrl.h
                                          int xs = -header.GetScroll();
C HeaderCtrl.cpp
                C Bar.cpp
# ArrayCtrl.h
                C MenuBar.cpp
                                          for(js = 0; js < column.GetCount(); js++) {</pre>
C ArrayCtrl.cpp
                C ToolBar.cpp
                                             int cw = header.GetTabWidth(js)
# DropChoice.h
                C ToolTip.cpp
                                              if ( ( xs + cw - vertgrid + (js == column.GetCount() - 1)) >= 0)
C DropBox.cpp
                # StatusBar.h
                                                  break;
C DropList.cpp
                C StatusBar.cpp
                                              xs += cw;
C DropPusher.cpp
               ▶ TabCtrl
                                          Color fc = Blend(SColorDisabled, SColorPaper);
C DropChoice.cpp
               # TabCtrl.h
                                          if(!IsNull(i))
                C TabCtrl.cpp
# StaticCtrl.h
                                              while(i < GetCount())</pre>
                > TreeCtrl
C Static.cpp
                                                   r.top = GetLineY(i) - sb;
# Splitter.h
                # TreeCtrl.h
                                                   if(r.top > size.cy) break;
C Splitter.cpp
                C TreeCtrl.cpp
                                                   r.bottom = r.top + GetLineCy(i);
☼ FrameSplitter.cpp   DlgColor
                                                   int x = xs;
# SliderCtrl.h
                # DlgColor.h
                                                   for(int j = js; j < column.GetCount(); j++) {</pre>
C SliderCtrl.cpp
                C DigColor.cpp
                                                        int cw = header.GetTabWidth(j);
# ColumnList.h
                ColorPopup.cpp
                                                        int cm = column[j].margin;
                ColorPusher.cpp
ColumnList.cpp
                                                        if(cm < 0)
# Progress.h
               FileSel
                                                            cm = header.Tab(j).GetMargin();
                # FileSel.h
Progress.cpp
                                                        if(x > size.cx) break;
# AKeys.h
                C FileList.cpp
                                                        r.left = x;
```

environment or users. Operating systems, such as Windows, macOS and such are usually a combination of C++, C#, assembly and some form of visual-based language. Games generally use C++ whilst web pages can use a plethora of available languages such as HTML, Java, Python and so on.

More general-purpose programming is used to create programs, apps, software or whatever else you want to call them. They're widely used across all hardware platforms and suit virtually every conceivable application. Some operate faster than others and some are easier to learn and use than others. Python is one such general-purpose language.

Python is what's known as a High-Level Language, in that it 'talks' to the hardware and operating system using a variety of arrays, variables, objects, arithmetic, subroutines, loops and countless more interactions. Whilst it's not as streamlined as a Low-Level Language, which can deal directly with memory addresses, call stacks and registers, its benefit is that it's universally accessible and easy to learn.

```
/file: Invoke.java
import java.lang.reflect.*;
class Invoke {
  public static void main( String [] args ) {
      class c = class.forName( args[0] );
      Method m = c.getMethod( args[1], new Class
      Object ret = m.invoke( null, null );
      System.out.println(
          "Invoked static method: " + args[1]
+ " of class: " + args[0]
+ " with no args\nResults: " + ret );
    } catch ( ClassNotFoundException e ) {
      // class.forName( ) can't find the class
    } catch ( NoSuchMethodException e2 ) {
      // that method doesn't exist
    } catch ( IllegalAccessException e3 ) {
      // we don't have permission to invoke that
      method
    } catch ( InvocationTargetException e4 ) {
      // an exception ocurred while invoking that
      System.out.println(
           'Method threw an: " + e4.
           getTargetException( ) );
  Java is a powerful
 language that's used in
 web pages, set-top boxes,
 TVs and even cars.
```

Python was created over twenty six years ago and has evolved to become an ideal beginner's language for learning how to program a computer. It's perfect for the hobbyist, enthusiast, student, teacher and those who simply need to create their own unique interaction between either themselves or a piece of external hardware and the computer itself.

Python is free to download, install and use and is available for Linux, Windows, macOS, MS-DOS, OS/2, BeOS, IBM i-series machines, and even RISC OS. It has been voted one of the top five programming languages in the world and is continually evolving ahead of the hardware and Internet development curve.

So to answer the question: why Python? Simply put, it's free, easy to learn, exceptionally powerful, universally accepted, effective and a superb learning and educational tool.

```
py=15
        FOR
                w=1 TO
                by=INT
                            (RND *28)
                bx = 0
        FOR d=1 TO 20
PRINT AT PX,PY;" U
PRINT AT bx,by;"o"
IF INKEY$="p" THEN
   90
  100
                                  THEN
  120
  130
              INKEY$="0" THEN LET PY=P
                         U
                       TO 100: NEXT
  135
             PY <2 THEN LET PY =2
PY >27 THEN LET PY =27
bx = bx +1
  180
         PRINT AT bx-1,by;
        IF (by-1) =py THEN LET s=s+1
PRINT AT 10,10; "score=";s
FOR v=1 TO 1000: NEXT v
  200
 210
 300 NEXT
Ø OK, Ø:1
```



BASIC was once the starter language that early 8-bit home computer users learned.

```
print(HANGMAN[0])
attempts = len(HANGMAN) - 1
while (attempts != 0 and "-" in word_guessed):
   print(("\nYou have {} attempts remaining").format(attempts))
    joined_word = "".join(word_guessed)
    print(joined word)
       player_guess = str(input("\nPlease select a letter between A-Z" + "\n> ")).
    except: # check valid inp
       print("That is not valid input. Please try again.")
       continue
       if not player_guess.isalpha(): # check the input is a letter. Also checks a
           print("That is not a letter. Please try again.")
           continue
        elif len(player_guess) > 1: # check the
           print("That is more than one letter. Please try again.")
            continue
       elif player_guess in guessed_letters: # check it letter hasn't been guessed
           print("You have already guessed that letter. Please try again.")
   guessed letters.append(player guess)
    for letter in range(len(chosen word)):
        if player guess == chosen word[letter]:
            word_guessed[letter] = player_guess # replace all letters in the chosen
   if player_guess not in chosen_word:
```



Python is a more modern take on BASIC, it's easy to learn and makes for an ideal beginner's programming language.