Отчет по форме обучения

30 ноября 2021 г. 23:02:43 Печать

Фамилия	Имя	Отчество	Дата рождения	Отделение	Код специальности	Специальность	Вид ограничения
Зенов	Артур	Рамазанович	06.10.2002	Экономическое	23.02.07	Техническое обслуживание и ремонт двигателей, систем и агрегатов автомобилей	Артрит
Чомаев	Рамазан	Юсуфович	11.11.2001	Экономическое	23.02.07	Техническое обслуживание и ремонт двигателей, систем и агрегатов автомобилей	Соматическое заболевание
Даюнов	Кемал	Русланович	27.08.2002	Экономическое	35.02.16	Эксплуатация и ремонт сельскохозяйственной техники и оборудования	Зрение
Чочуев	Акимат	Хамитович	21.02.2003	Экономическое	35.02.16	Эксплуатация и ремонт сельскохозяйственной техники и оборудования	Неврологический/гидроцефалический/гипертензивный синдром
Байтонова	Дина	Рашитовна	13.11.2000	Экономическое	21.02.05	Земельно-имущественные отношения	Зрение

Рисунок 11. Отчет по форме обучения в режиме представления отчета

Таким образом, разработана база данных для лиц с ограниченными возможностями, созданы таблицы, запросы, формы и отчеты в СУБД Access.

Морозова Н.В., Мамчуев А.К., Бойко Э.А., Байрамкулов И.Х.

Выбор языка программирования для начинающего программиста

*Северо-кавказская государственная академия
(Россия, Черкесск)*

doi: 10.18411/trnio-12-2021-99

Аннотация

В данной статье рассмотрены два языка программирования – это Pascal и Python, приведены особенности этих языков. Даны рекомендации о выборе первого или базового языка для обучения программированию новичкам.

Ключевые слова: Python, Pascal, язык программирования, IDE, библиотеки, GUI.

Abstract

This article discusses two programming languages - Pascal and Python, the features of these languages are given. Recommendations are given on the choice of the first or basic language for teaching programming to beginners.

Keywords: Python, Pascal, programming language, IDE, libraries, GUI.

Python

Python – простой в освоении и мощнейший язык программирования, по сравнению с другими языками. Он предоставляет действительно высокоуровневые структуры данных, так же простой, но действительно подход к ООП. Его стильный синтаксис и динамическая типизация вместе с тем, делают его безупречным языком для написания сценариев и резвой разработкой приложений в разных областях в большинстве площадоков.

Особенности Python

1. Простота.

Python – обычный и минималистичный язык. Чтение неплохой программы на Python чрезвычайно напоминает чтение британского текста, довольно

серьезного. Данная псевдо-кодовая природа Python является одной из его самых мощных сторон. Она позволяет сконцентрироваться на решении задачи, но не на самом языке. При написании программы на Python никогда не придется отвлекаться на такие низкоуровневые детали, как управление памятью, используемой вашей программой, и т. п.

2. Обширные библиотеки.

Стандартная библиотека Python просто огромна. Она может помочь в решении самых разнообразных задач, связанных с использованием регулярных выражений, генерированием документации, проверкой блоков кода, распараллеливанием процессов, базами данных, веб-браузерами, CGI, FTP, электронной почтой, XML, XML-RPC, HTML, WAV файлами,

криптографией, GUI (графическим интерфейсом пользователя) и другими системно зависимыми вещами. Всё это доступно абсолютно везде, где установлен Python. В этом заключается философия Python «Всё включено». Кроме стандартной библиотеки, существует множество других высококачественных библиотек, которые можно найти в каталоге пакетов Python.

3. Расширяемый и встраиваемый.

Если вы хотите, чтобы часть программы работала быстрее или вам надо скрыть часть алгоритма, то можете написать эту часть программы на C или C++, а затем вызвать её из программы Python. Python можно встраивать в программы на C/C++, что предоставляет возможности написания сценариев их пользователям.

4. Большой выбор IDE.

На Python огромное количество IDE на любой цвет и вкус. Например: PyCharm, VS Code, Visual Studio, Jupyter, Spyder и т.д.

Pascal

Pascal — один из более узнаваемых языков программирования, употребляется для обучения программированию в старших классах и на первых курсах высших учебных заведений, является основой для ряда остальных языков. Чертами языка являются строгая типизация и наличие средств структурного (процедурного) программирования. Паскаль был одним из первых таковых языков. Как считает Никлаус Вирт, язык должен содействовать дисциплинированному программированию, потому, наряду со серьезной типизацией, в Паскале сведены к минимуму вероятные синтаксические неоднозначности, а сам синтаксис автор постарался сделать интуитивно понятным даже при первом знакомстве с языком.

Особенности Pascal

1. Дисциплинирует.

Никлаус Вирт считает, что языки должны дисциплинировать программирование.

2. Pascal родственник C++.

После изучения Pascal легко перейти на C++, из-за схожести синтаксиса. Пример программы приведен на рисунке 1.

```
var x, a, b: integer;
begin
  readln(x);
  a:=0; b:=0;
  while x>0 do
  begin
    a:=a + 1;
    if b < (x mod 8)
    then
      b:=x mod 8;
    x:=x div 8;
  end;
  writeln(a); write(b);
end.
```

Pascal

```
#include <iostream>
using namespace std;
int main()
{
  int x, a, b;
  cin >> x;
  a=0; b=0;
  while (x>0){
    a = a+1;
    if (b < (x%8)){
      b = x%8;
    }
    x = x/8;
  }
  cout << a << endl << b << endl;
}
```

C++

Рисунок 1. Пример синтаксиса языка программирования Pascal C++

Одно из отличий этих двух примеров, в том что в C++ мы используем { }, а в Pascal to-do; в Pascal мы используем div или mod, а в C++ / или %.

Сравнение

У Python более простой синтаксис, в отличии от Pascal и других языков. Приведу в пример задачу: «Составить программу, выводящую на экран расписание учителя (номер урока, время начала урока, класс, предмет, номер кабинета)». Код программы представлен на рисунках 2-3.

```

type
  tovar = record
    nu: integer;
    vn: string;
    kl: string;
    pr: string;
    nk: integer;
  end;
var
  mas: array[1..10] of tovar;
  n, i: integer;

procedure input;
begin
  writeln('Запись №', i);
  write('Номер урока: ');
  readln(mas[i].nu);
  write('Время начала: ');
  readln(mas[i].vn);
  write('Класс: ');
  readln(mas[i].kl);
  write('Предмет ');
  readln(mas[i].pr);
  write('Номер кабинета ');
  readln(mas[i].nk);
  writeln;
end;

procedure output;
begin
  writeln(mas[i].nu:13, ' | ', mas[i].vn:15, ' | ', mas[i].kl:10, ' | ',
  mas[i].pr:15, ' | ', mas[i].nk:15, ' | ');
end;

begin
  writeln('Количество записей:');
  readln(n);

  writeln;
  writeln('Введите данные об уроках ');
  for i:=1 to n do
    input;

  writeln;
  writeln('Расписание учителя');
  writeln('Номер урока':13, ' | ', 'Время начала':15, ' | ',
  'Класс':10, ' | ', 'Предмет':15, ' | ', 'Номер кабинета':15, ' | ');
  for i:=1 to n do
    output;
end.

```

Рисунок 2. Пример кода на языке программирования Pascal

```

import sys
from prettytable import PrettyTable

i = 0
#Создание списков
teacherSchedule = []
myTable = PrettyTable()

#Ввод значений
numberOfLessons = str(input("Сколько уроков ? "))
intNumberOfLessons = int(numberOfLessons)

while i < intNumberOfLessons:
    lessonNumber = str(input("Номер урока: "))
    startTime = str(input("Время начало: "))
    Class = str(input("Класс: "))
    item = str(input("Предмет: "))
    roomNumber = str(input("Номер кабинета: "))

    #Создание таблицы
    myTable.field_names = ["Номер урока", "Время начало",
    "Класс", "Предмет", "Номер кабинета"]
    myTable.add_row([lessonNumber, startTime, Class, item, roomNumber])
    i = i + 1
print(myTable)

```

Рисунок 3. Пример кода на языке программирования Python

На этих двух скриншотах мы видим, что код на Python'е более простой и понятный, чем на Pascal. Программирование на Pascal'е занимает больше времени, чем на Python'е. Например, чтобы запросить у пользователя ввод значения для переменной на Python'е мы пишем:

```
lessonNumber = str(input("Номер урока: "))
```

В то время как на Pascal'е это занимает две строки:

```
write('Номер урока:');
```

```
readln(mas[i].nu);
```

Конкретно в этом случае на Pascal мы сохраняем номер урока сразу в массив, а в других случаях пришлось бы описывать переменные. На Python'е нам не нужно каждый раз описывать переменную.

Отступы и точка-запятая

В языке Pascal весь код можно написать в один ряд и обязательно должны быть точки-запятые. В Python'е все не так. В Python'е точки запятые не ставятся и обязательно должны соблюдаться отступы.

Библиотеки

Стандартное количество библиотек Python'а обширно, но кроме стандартной библиотеки в Python'е существует множество других высококачественных библиотек, которые можно найти в каталоге пакетов Python, что нельзя сказать то же самое о Pascal'е. В Pascal есть библиотеки, но они уступают библиотекам Python своей масштабностью. Там нет некоторых библиотек, которые есть на Python, например, графической библиотеки. В Pascal графической библиотеки нет, в то время как в Python их 13 штук: TKinter, PyQt, Flex, CEF Python, Dabo, Kivy, Pyfroms, PyGObject, PySide, PyGUI, libavg, PyGTK, wxPython.

Вакансии

Сейчас Pascal-разработчику найти работу практически невозможно, в отличие от Python-разработчика. В настоящее время один из самых востребованных разработчиков Python-разработчики. В настоящее время Pascal используется в основном только для обучения программированию в старших классах и на первых курсах вузов, в то время как Python используется в анализе данных, машинном обучении, веб-разработке и в других сферах.

Вывод

В данной статье на основе анализа особенностей двух языков программирования, рекомендуем начинающим разработчикам в качестве базового языка использовать Python.

1. <https://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.02.pdf>
2. <https://sites.google.com/site/paskalpro/osnovatel-azyka>
3. [https://ru.wikipedia.org/wiki/Паскаль_\(язык_программирования\)](https://ru.wikipedia.org/wiki/Паскаль_(язык_программирования))

Морозова Н.В., Махнов В.А.

Создание базы данных в СУБД Firebird с использованием среды разработки IBExpert

*Северо-кавказская государственная академия
(Россия, Черкесск)*

doi: 10.18411/trnio-12-2021-100

Аннотация

В данной статье рассмотрены начальные этапы создания базы данных в среде разработки IBExpert на основе СУБД Firebird. В ней на конкретных примерах приведён план действий, который поможет начинающим разработчикам проводить начальную настройку и простейшие операции с базами данных на первых этапах освоения рабочей среды IBExpert.

Ключевые слова: таблицы, база данных, скрипты СУБД, FireBird, IBExpert, SQL.