# Python课程第一节课教案

## 课程基本信息

|  |  |
| --- | --- |
| 项目 | 内容 |
| 课程名称 | Python编程入门 |
| 课时 | 第1课（共12课） |
| 授课时长 | 90分钟 |
| 适用对象 | 编程初学者 |
| 教学目标 | 了解Python基础语法与开发环境 |

## 教学目标

* 了解Python语言的特点与应用场景
* 安装并配置Python开发环境
* 编写并运行第一个Python程序
* 理解变量、数据类型与基本输入输出

## 课程内容大纲

### 1. Python简介

Python是一种高级、解释型、通用编程语言，具有以下特点：

* \*\*简洁易读\*\*：语法接近自然语言
* \*\*跨平台\*\*：支持Windows、macOS、Linux等系统
* \*\*丰富的库\*\*：广泛应用于Web开发、数据分析、人工智能等领域



Python Logo

\*\*提示\*\*：Python之父是Guido van Rossum，于1991年首次发布。

### 2. 开发环境搭建

#### 所需工具

* Python解释器（推荐3.8+版本）
* 代码编辑器（如VS Code、PyCharm、Thonny等）

#### 安装步骤（以Windows为例）

1. 访问 [https://www.python.org/downloads/](https://www.python.org/downloads/)
2. 下载最新稳定版安装包
3. 运行安装程序，\*\*务必勾选“Add Python to PATH”\*\*
4. 验证安装：打开命令提示符，输入 `python --version`

### 3. 第一个Python程序

# hello.py

print("Hello, World!")

name = input("请输入你的名字：")

print(f"你好，{name}！欢迎学习Python！")

**运行方式**：

* 在终端中执行：`python hello.py`

### 4. 基础概念讲解

#### 常见数据类型

|  |  |  |
| --- | --- | --- |
| 数据类型 | 示例 | 说明 |
| `int` | `42` | 整数 |
| `float` | `3.14` | 浮点数 |
| `str` | `"Python"` | 字符串（用引号包围） |
| `bool` | `True`, `False` | 布尔值 |

#### 变量命名规则

* 以字母或下划线开头
* 只能包含字母、数字和下划线
* 区分大小写（`name` ≠ `Name`）
* 避免使用Python关键字（如 `if`, `for`, `class`）

### 5. 课堂练习

1. 编写程序输出你的姓名和年龄
2. 编写程序计算两个数的和
3. 尝试修改字符串中的内容并输出

## 课后作业

* 安装Python并成功运行“Hello, World!”程序
* 阅读Python官方入门教程前两章
* 思考：Python与其他语言（如Java、C++）相比有何优势？

## 教学资源

* [Python官方文档](https://docs.python.org/zh-cn/3/)
* [菜鸟教程 - Python3](https://www.runoob.com/python3/)
* 推荐书籍：《Python编程：从入门到实践》

---

\*\*教师备注\*\*：本节课重在激发兴趣，避免过度强调语法细节。鼓励学生动手尝试，及时解答环境配置问题。