

# 第四集： Bash 脚本入门

## 课程目标

本课程将带你入门 Bash 脚本的编写和执行。通过学习本集内容，你将学会如何创建和运行 Bash 脚本，并理解基础语法，如变量、条件语句和循环的使用。

### 1. 创建第一个 Bash 脚本

- **功能：** 编写一个简单的 Bash 脚本并运行它。
- **步骤：**
  - 使用 `nano` 或 `vim` 打开一个新文件，例如 `script.sh`。
  - 在文件的第一行添加 `#!/bin/bash`，声明脚本的解释器。
  - 在脚本中添加以下内容：

```
echo "Hello, Bash!"
```

- 保存并关闭文件。
- 通过以下命令为脚本添加可执行权限：

```
chmod +x script.sh
```

- 使用以下命令运行脚本：

```
./script.sh
```

### 2. 脚本中的基础语法

- **变量：**
  - 变量的定义和使用：

```
name="Bash User"
echo "Hello, $name"
```

- **实操：**创建一个脚本，定义变量并输出其值。
- **条件语句（if-else 结构）：**
  - 条件判断的基本结构：

```
if [ condition ]; then
    # 代码块
else
    # 另一个代码块
fi
```

```
#!/bin/bash

# 读取用户输入的数字
echo "请输入一个数字： "
read number

# 判断数字的类型
if [ $number -gt 0 ]; then
    echo "这个数字是正数。"
elif [ $number -lt 0 ]; then
    echo "这个数字是负数。"
else
    echo "这个数字是零。"
fi
```

- **示例：**判断一个数字是否为0并输出相应信息。
- **实操：**编写一个脚本进行简单的条件判断。
- **循环：**

- for 和 while 循环的基本用法：

- for 循环示例：

```
for i in {1..5}; do
    echo "Number: $i"
done
```

- while 循环示例：

```
count=1
while [ $count -le 5 ]; do
    echo "Count: $count"
    count=$((count + 1))
done
```

### 3. 实操任务

- **任务 1：** 创建一个名为 `hello.sh` 的脚本，并输出 "Hello, World!"。
- **任务 2：** 为 `hello.sh` 添加可执行权限并运行，查看输出结果。
- **任务 3：** 创建一个带有变量和条件判断的脚本，输出变量的值并判断是否满足特定条件。

### 小结

本集课程中，我们学习了如何创建和运行 Bash 脚本，以及基础语法的应用。通过实操示例，我们初步掌握了编写脚本的基础方法。

**下节预告：** 在下集课程中，我们将学习 Bash 中的管道与重定向，深入了解如何利用命令的输出与输入进行操作。