

原文在[这里](#)

我正在参加 Joel Solon 讲授的“使用 InterSystems Objects 和 SQL 进行开发”课程。课程非常好，我将在这里分享一些从培训中总结的提示。

1. 您可以创建持久类（在数据库中具有对应表的类，用于保持类属性）。

2. 持久类示例：

[illegible]

3. 扩展 %Persistent 时，您将获得 %New() 以在内存中创建新实例，获得 %Save() 以保存到数据库，获得 %Id() 以获取该实例在数据库中的唯一 ID，以及获得 %OpenId() 以使用数据库值加载实例。

4. 持久类允许您调用 %Deleteid() 以从数据库中删除一个实例，调用 %DeleteExtent() 以删除所有保存的对象（没有 where 时删除！），调用 %ValidateObject() 以验证保存前传递的数据（验证是否必需、大小等）。

5. 持久类具有 `%IsModified()` 和 `%Reload()`，前者用于检查内存中的数据变化（参见评论中 joel 的提示），后者用于获取这些变化。

6. 要在尝试 %Save() 或 %Delete() 时获得可能的错误，请执行：set status = person.%Save(), write status。
如果保存成功，将返回 1。

7. 我们可以使用 `do $system.Status.DisplayError(status)` 查看错误详细信息。

8. 要调用持久类方法，请执行：`##class(dc.Person).%Save()`。

9. 要调用持久实例方法，执行：`..Method()`。引用属性也一样，请执行：`write ..Name`。

10. 要从程序或终端内存中移除对象或变量，请使用 `kill person` 或 `set person = ""`。如果仅使用 `kill`，将从内存中移除所有引用（不是从数据库中移除，在数据库中使用 `killextent`）。

11. 如果需要通过实用工具方法填充测试数据，请使用 %Populate 扩展持久类，然后调用 Populate(行数)方法。

12. 您可以通过继承%SerialObject（不具有 ID 的持久类，因为它必须与另一个持久类相连）创建嵌入式类。示例：

```
Class dc.Contact Extends %SerialObject
{
    Property Phone As %String;

    Property Email As %String;
}
```

13. 此序列将成为您的持久类的一个属性：

```
Class dc.Person extends (%Persistent)
{
    Property Name As %String;

    Property BirthDate As %Date;

    Property Contact As dc.Contact;
}
```

14. 在 IRIS 数据库中，将仅创建一个具有 Contact 属性的 Person 表。

15. 您可以创建索引来获得唯一性或调整查询。示例：Index NameIndex On Name [Unique]。

16. 创建索引时，如果表不为空，则需要在管理门户中重建索引。

17. 要创建构造函数方法，请重写 %OnNew()。这是在调用 %New() 时调用的回调方法。还有其他回调方法。

18. IRIS 对 JSON 有很好的支持。您可以通过调用 set name = {}.%FromJSON("{\"Name\":\"Yuri\"}") 将 JSON 加载到对象。

19. 您可以从对象执行编写 JSON：name.%ToJSON()。

20. IRIS 和 Caché 中存在 JSON 数组（感谢 [@Robert C. Cemper](#) 提醒），但是只在 IRIS 中我们才有 JSON 的格式化程序和 zwrite 写入 JSON。

明天我会发布第 3 天的总结。

PS：这是一份总结，课程中实际教授了更多内容。

[#提示和技巧](#) [#InterSystems IRIS](#)

源

URL:

<https://cn.community.intersystems.com/post/%E7%AC%AC-2-%E5%A4%A9%E5%BC%9A%E4%BD%BF%E7%94%A8-intersystems-objects-%E5%92%8C-sql-%E8%BF%9B%E8%A1%8C%E5%BC%80%E5%8F%91>