

文章

[姚鑫](#) · 十月 24, 2021 阅读大约需 4 分钟

第五十五章 SQL命令 INSERT（四）

第五十五章 SQL命令 INSERT（四）

嵌入式SQL和动态SQL示例

下面的嵌入式SQL示例创建一个新表SQLUser.MyKids。下面的示例使用INSERT用数据填充此表。在插入示例之后，提供了一个删除SQLUser.MyKids的示例。

```
ClassMethod Insert2()
{
    &sql(
        CREATE TABLE SQLUser.MyKids
        (
            KidName VARCHAR(16) UNIQUE NOT NULL,
            KidDOB INTEGER NOT NULL,
            KidPetName VARCHAR(16) DEFAULT 'no pet'
        )
    )
    if SQLCODE=0 {
        w !,"???"
    } elseif SQLCODE=-201 {
        w !,"???"
        q
    } else {
        w !,"CREATE TABLE???SQLCODE=",SQLCODE
    }
}
```

下面的嵌入式SQL示例插入具有两个字段值的行(第三个字段KidPetName采用默认值)。请注意，表架构名称由#SQL Compile Path宏指令作为架构搜索路径提供：

```
ClassMethod Insert3()
{
    #SQLCompile Path = Sample
    n SQLCODE,%ROWCOUNT,%ROWID
    &sql(
        INSERT INTO MyKids
        (
            KidName, KidDOB
        )
        VALUES
        (
            'Molly', 60000
        )
    )
    if SQLCODE = 0 {
```

```

        w !, "?????"
        w !, "Row count=", %ROWCOUNT
        w !, "Row ID=", %ROWID
        q
    } elseif SQLCODE = -119 {
        w !, "????????", !
        w %msg, !
        q
    } else {
        w !, "?????SQLCODE=", SQLCODE
    }
}

```

下面的嵌入式SQL示例使用表的列顺序插入具有三个字段值的行：

```

ClassMethod Insert4()
{
    n SQLCODE, %ROWCOUNT, %ROWID
    &sql(
        INSERT INTO SQLUser.MyKids VALUES ('Josie', '40100', 'Fido')
    )
    if SQLCODE = 0 {
        w !, "?????"
        w !, "Row count=", %ROWCOUNT
        w !, "Row ID=", %ROWID
        q
    } elseif SQLCODE = -119 {
        w !, "????????", !
        w %msg, !
        q
    } else {
        w !, "?????SQLCODE=", SQLCODE
    }
}

```

下面的嵌入式SQL示例使用主机变量插入具有两个字段值的行。这里使用的插入语法指定了COLUMN=VALUE对：

```

ClassMethod Insert5()
{
    #SQLCompile Path=Sample
    n SQLCODE, %ROWCOUNT, %ROWID
    s x = "Sam"
    s y = "57555"
    &sql(
        INSERT INTO MyKids
        SET
            KidName = :x, KidDOB = :y
    )
    if SQLCODE = 0 {
        w !, "?????"
        w !, "Row count=", %ROWCOUNT
        w !, "Row ID=", %ROWID
        q
    } elseif SQLCODE = -119 {

```

```

        w !, "???????", !
        w %msg, !
    q
} else {
    w !, "?????SQLCODE=", SQLCODE
}
}

```

下面的嵌入式SQL示例使用主机变量数组插入具有三个字段值的行。数组元素按列顺序编号。请注意，用户提供的数组值以myarray(2)开头；第一个数组元素对应于RowID列，该列是自动提供的，不能由用户定义：

```

ClassMethod Insert6()
{
    #SQLCompile Path=Sample
    n SQLCODE, %ROWCOUNT, %ROWID
    s myarray(2) = "Deborah"
    s myarray(3) = 60200
    s myarray(4) = "Bowie"
    &sql(
        INSERT INTO MyKids VALUES :myarray()
    )
    if SQLCODE = 0 {
        w !, "???"
        w !, "Row count=", %ROWCOUNT
        w !, "Row ID=", %ROWID
        q
    } elseif SQLCODE = -119 {
        w !, "???????", !
        w %msg, !
        q
    } else {
        w !, "?????SQLCODE=", SQLCODE
    }
}

```

下面的动态SQL示例使用%SQL.Statement类插入具有三个字段值的行。请注意，表架构名称在%New()方法中作为架构搜索路径提供：

```

ClassMethod Insert7()
{
    s x = "Noah"
    s y = "61000"
    s z = "Luna"
    s sqltext = "INSERT INTO MyKids (KidName,KidDOB,KidPetName) VALUES (?, ?, ?)"
    s tStatement = ##class(%SQL.Statement).%New(0, "Sample")
    s qStatus = tStatement.%Prepare(sqltext)
    if qStatus '= 1 {
        w "%Prepare failed:"
        d $System.Status.DisplayError(qStatus)
        q
    }
    s rtn = tStatement.%Execute(x, y, z)
    if rtn.%SQLCODE = 0 {
        w !, "???"
        w !, "Row count=", rtn.%ROWCOUNT
    }
}

```

```

        w !, "Row ID=", rtn.%ROWID
    } elseif rtn.%SQLCODE = -119 {
        w !, "???????", !, rtn.%Message
        q
    } else {
        w !, "?????SQLCODE=", rtn.%SQLCODE
    }
}

```

下面的嵌入式SQL示例显示插入的记录，然后删除SQLUser.MyKids表：

```

ClassMethod Insert8()
{
    s myquery = "SELECT * FROM SQLUser.MyKids"
    s tStatement = ##class(%SQL.Statement).%New()
    s qStatus = tStatement.%Prepare(myquery)
    if qStatus '= 1 {
        w "%Prepare failed:"
        d $System.Status.DisplayError(qStatus)
        q
    }
    s rset = tStatement.%Execute()
    d rset.%Display()
    w !, "End of data"
    &sql(DROP TABLE SQLUser.MyKids)
    if SQLCODE = 0 {
        w !, "Deleted table"
        q
    } else {
        w !, "Table delete failed, SQLCODE=", SQLCODE
    }
}

```

```
DHC-APP>d ##class(PHA.TEST.SQLCommand).Insert8()
```

```

KidName KidDOB KidPetName
Molly 60000 no pet
Josie 40100 Fido
Sam 57555 no pet
Deborah 60200 Bowie
Noah 61000 Luna

```

```

5 Rows(s) Affected
End of data
Deleted table

```

下面的嵌入式SQL示例演示了主机变量数组的使用。请注意，对于主机变量数组，可以使用带有未指定最后一个下标的动态本地数组来传递要在运行时插入的值数组。例如：

```

ClassMethod Insert9()
{
    n SQLCODE,%ROWCOUNT,%ROWID
    &sql(INSERT INTO Sample.Employee VALUES :emp('profile',))
    w !, "SQL Error code: ", SQLCODE, " Row Count: ", %ROWCOUNT
}

```

使插入的“Employee”行中的每个字段设置为：

```
emp("profile", col)
```

其中，“col”是Sample.Employee表中字段的列号。

下面的示例显示如何将SELECT查询的结果用作INSERT语句的数据输入，从而为多行提供数据：

```
INSERT INTO StudentRoster (NAME,GPA,ID_NUM)
SELECT FullName,GradeAvg,ID
FROM temp WHERE SchoolYear = '2004'
```

#SQL #Caché

URL:

<https://cn.community.intersystems.com/post/%E7%AC%AC%E4%BA%94%E5%8D%81%E4%BA%94%E7%AB%A0-sql%E5%91%BD%E4%BB%A4-insert%EF%BC%88%E5%9B%9B%E7%BC%89>