

Jenkins

- description :
- author :
- email : iyyeo@repia.com
- lastupdate : 2021-09-15

```
import
com.intelli
j.database.
model.DasTa
ble
import
com.intelli
j.database.
model.Objec
tKind
import
com.intelli
j.database.
util.Case
import
com.intelli
j.database.
util.DasUti
l
import
com.intelli
j.psi.codeS
tyle.NameUt
il

import
javax.swing
.*

/**
 * @author
shirohoo
 * @link
https://git
hub.com/shi
rohoo/creat
e-
automation-
jpa-entity
 * @param
packageName
,
primaryKey
*
 * <pre>
 *
 *      this
script's
```

```
default
primary key
strategy is
@GeneratedV
alue(strate
gy =
GenerationT
ype.IDENTIT
Y)
* and
specialized
in
Microsoft
SQL Server
* and
finally
implemented
Serializable
e so
recommend
that create
serial
version UID
*
*
first.
enter your
project
package
name. for
example:
* >
com.intelli
J.psi
*
*
second.
enter
primary key
column name
of target
database
table.
* this
script is
convert
input to
camel case.
for example
1:
* >
```

```
table
primary key
column name
= MEMBER_ID
* >
enter
primary key
= memberId
*
*
example 2:
* >
table
primary key
column name
= ID
* >
enter
primary key
= id
*
* </pre>
*/
```

```
columnType
= [
(~/(?i)bigi
nt/)
: "Long",
(~/(?i)int/
)
:
"Integer",
(~/(?i)bit/
)
:
"Boolean",
(~/(?i)deci
mal/)
:
"BigDecimal",
(~/(?i)floa
t|double|re
al/) :
"Double",
(~/(?i)date
time|timest
amp/):
"LocalDateT
ime",
```

```
(~/(?i)time
/)
:
"LocalTime"
,
(~/(?i)date
/)
:
"LocalDate"
,
(~/(?i)nvar
char/)
:
"nvarchar",
(~/(?i)varc
har/)
:
"varchar",
(~/(?i)char
/)
: "String",
(~/(?i)text
/)
: "String"
]
def input =
{
    JFrame
    jframe =
new
JFrame()
    String
    answer =
JOptionPane
.showInputD
ialog(jfram
e, it)
jframe.disp
ose()
    answer
}
packageName
=
input("Ente
r your
package
name")
primaryKey
=
input("Ente
r column
```

```
name of
primary
key")
FILES.choos
eDirectoryA
ndSave("Cho
ose
directory",
"Choose
where to
store
generated
files") {
dir ->
SELECTION.f
ilter {
    it
instanceof
DasTable &&
it.getKind(
) ==
ObjectKind.
TABLE
    }.each
{
generate(it
, dir)
}
}

def
generate(ta
ble, dir) {
    def
tableName =
table.getNa
me()
    def
className =
convertFiel
dName(table
Name +
"Entity",
true)
    def
fields =
categorizeF
ields(table
)
    new
File(dir,
```

```
className +  
".java").withPrintWriter {  
    out  
->  
generate(out,  
tableName,  
className,  
fields)  
}
```

```
def  
generate(out,  
tableName,  
className,  
fields) {  
out.println  
"package  
$packageName;"  
out.println  
""  
out.println  
"import  
javax.persistence.*;"  
out.println  
"import  
java.time.LocalDate;"  
out.println  
"import  
lombok.Data;"  
out.println  
""  
out.println  
"@Entity"  
out.println  
"@Data"  
out.println  
"@Table(name =  
\"$tableName\")"  
out.println
```

```
"public
class
$class_name"
out.println
 "{"
fields.each
 () {
     if
 (it.annos
 != "") {
 out.println
 "
 ${it.annos}
 "
     }
     if
 (it.name ==
 primaryKey)
 {
 out.println
 "    @Id"
 out.println
 "
 @GeneratedValue
 alue(strategy =
 GenerationType.
 IDENTIT
 Y)"
     }
     if
 (it.type ==
 'nvarchar')
 {
 out.println
 "
 @Nationaliz
 ed"
 out.println
 "
 @Column(nam
 e = \"${it.
 colName}\")
 "
 out.println
 "    private
 String
 ${it.name};
 "
     }
 else if
```

```
(it.type ==
'varchar')
{
out.println
"
@Column(name =
\"${it.colName}\")
out.println
"    private
String
${it.name};
"
}
else {
out.println
"
@Column(name =
\"${it.colName}\")
out.println
"    private
${it.type}
${it.name};
"
}
out.println
""
}
out.println
"}"
}

def
categorizeFields(table) {
DasUtil.getColumns(table).reduce(
[]) {
fields, col
->
def
spec =
Case.LOWER.apply(col.getDataType().getSpecif
```

```
ication()
    def
    typeStr =
    columnType.
    find {
    p, t ->
    p.matcher(s
    pec).find()
    }.value
    fields +=
    [[
    colName:
    col.getName
    (),
    name :
    convertFiel
    dName(col.g
    etName(),
    false),
    type :
    typeStr,
    annos :
    ""]]
    }
}

def
convertFiel
dName(str,
capitalize)
{
    def s =
    NameUtil.sp
    litNameInto
    Words(str)
    .collect {
    Case.LOWER.
    apply(it).c
    apitalize()
    }
    .join("")
    .replaceAll
    (/^[^p{java
    JavaIdentif
    ierPart}[_]
    ]/, "_")
    capitalize
    ||
    s.length()
    == 1 ? s :
    Case.LOWER.
```

```
apply(s[0])  
+ s[1..-1]  
}
```

Tip

Troubleshooting

Ref

, [Entity](#), [Intellij](#), , ,

From:

<http://rwiki.repia.com/> -

. - 2023.12

Permanent link:

<http://rwiki.repia.com/doku.php?id=wiki:user:iyyeo:entity&rev=1631664974>



Last update: **2022/03/10 19:52**