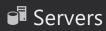
Project / Servers Hoop Testing





Local_Development

Project / Servers / Local_Development	Hoop Testing
■ Local_Development	
Databases data_db static_db	
Users dataCollector@% databaseManager@% reportGenerator@% Properties	
Name	Value
Product	MySQL
Version	8.3.0
Version Comment	MySQL Community Server - GPL
Collation	utf8mb4_0900_ai_ci







static_db



Properties

Name	Value
Collation	utf8mb4_0900_ai_ci

Object Types 4





Procedures



7 Triggers



Objects 3

Name	Description
data	Data Recollected from PLC
sample	Test Sample information
specimen	Test specimen



Data Recollected from PLC

Properties

Name	Value
Engine	InnoDB
Auto Increment	3887
Average Row Length	58
Charset	utf8mb4
Collation	utf8mb4_general_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	11/6/2024 11:05:27
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	specimen	INT		11		True	False	False	True	False		False	False	
	pressure	DOUBLE	0			False	False	False	True	False		False	False	
	temperature	DOUBLE	0			False	False	False	True	False		False	False	
	createdAt	DATETIME		4		False	False	False	False	False	CURRENT_TIMESTAMP(4)	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
₩		id	True	None	0
	IDX_test_data_test_specimen	specimen	False	None	0

Foreign Keys

Name	Columns	Delete Rule	Update Rule

FK_test_data_test_specimen_id id N/S N/S

SQL Script

```
CREATE TABLE data (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   specimen int UNSIGNED NOT NULL,
   pressure double NOT NULL,
   temperature double NOT NULL,
   createdAt datetime(4) DEFAULT CURRENT_TIMESTAMP(4),
   PRIMARY KEY (id)
)

ENGINE = INNODB,
   AVG_ROW_LENGTH = 58,
   CHARACTER SET utf8mb4,
   COLLATE utf8mb4_general_ci,
   COMMENT = 'Data Recollected from PLC';

ALTER TABLE data
   ADD INDEX IDX_test_data_test_specimen (specimen);

ALTER TABLE data
   ADD CONSTRAINT FK_test_data_test_specimen_id FOREIGN KEY (specimen)
   REFERENCES specimen (id);
```

Depends On 1



Used By 6



5 DeleteTest



selectTestData



selectTestLimit



Test Sample information

Properties

Name	Value
Engine	InnoDB
Auto Increment	50
Average Row Length	16384
Charset	utf8mb4
Collation	utf8mb4_general_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	11/6/2024 11:05:27
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	standard	VARCHAR	25			False	False	False	True	False		False	False	
	material	VARCHAR	10			False	False	False	True	False		False	False	
	specification	VARCHAR	10			False	False	False	True	False		False	False	
	diamreal	INT		11		True	False	False	True	False		False	False	
	diamnom	INT		11		True	False	False	True	False		False	False	
	wallthick	INT		11		True	False	False	True	False		False	False	
	lentotal	INT		11		True	False	False	True	False		False	False	
	lenfree	INT		11		True	False	False	True	False		False	False	
	targettemp	INT		11		False	False	False	True	False		False	False	
	targetpressure	INT		11		False	False	False	True	False		False	False	
	condPeriod	VARCHAR	15			False	False	False	True	False		False	False	

createdAt	DATETIME	0	False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
updatedAt	DATETIME	0	False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
⊡		id	True	None	0

Triggers

Name	Туре	Event	Security	Definer
NewSample	BEFORE	INSERT	Definer	root

SQL Script

```
CREATE TABLE sample (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   standard varchar(25) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
   material varchar(10) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
   specification varchar(10) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
   diamreal int UNSIGNED NOT NULL,
   diamnom int UNSIGNED NOT NULL,
   wallthick int UNSIGNED NOT NULL,
   lentotal int UNSIGNED NOT NULL,
   lentree int UNSIGNED NOT NULL,
   targettemp int NOT NULL,
   targettemp int NOT NULL,
   condPeriod varchar(15) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
   createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   updatedAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
   PRIMARY KEY (id)
)

ENGINE = INNODB,
   AVG_ROW_LENGTH = 16384,
   CHARACTER SET utf8mb4,
   COLLATE utf8mb4_general_ci,
   COMMENT = 'Test Sample information';
```

Depends On

No items found

Used By 12

- insertSample
- insertSpecimen
- selectSample
- selectSampleID
- selectCompareTests
- selectTest
- selectTests
- NewSample
- specimen
- <u>≡</u> getSpecimenData
- selectTestSample
- ✓ DeleteTestSample

specimen

Description

Test specimen

Properties

Name	Value
Engine	InnoDB
Auto Increment	136
Average Row Length	910
Charset	utf8mb4
Collation	utf8mb4_general_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	11/6/2024 11:05:27
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	sample	INT		11		True	False	False	True	False		False	False	
	operator	VARCHAR	40			False	False	False	True	False	'STEL S.A.'	False	False	
	enviroment	VARCHAR	20			False	False	False	False	False		False	False	
	testName	VARCHAR	120			False	False	False	False	False		False	False	
	endCap	VARCHAR	10			False	False	False	False	False		False	False	
	failText	VARCHAR	200			False	False	False	True	False	'Sin Fallas'	False	False	
	remark	VARCHAR	200			False	False	False	True	False	'Sin Observaciones'	False	False	
	createdAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
	updatedAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
ದಾ		id	True	None	0
	FK_test_specimen_test_sample_id	sample	False	None	0

Triggers

Name	Туре	Event	Security	Definer
DeleteTest	BEFORE	DELETE	Definer	root
DeleteTestSample	AFTER	DELETE	Definer	root

Foreign Keys

Name	Columns	Delete Rule	Update Rule
FK_test_specimen_test_sample_id	id	N/S	N/S

SQL Script

```
CREATE TABLE specimen (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   sample int UNSIGNED NOT NULL,
   operator varchar(40) CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT 'STEL S.A.',
   enviroment varchar(20) CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci DEFAULT '',
   testName varchar(120) CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci DEFAULT '',
   endCap varchar(10) CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci DEFAULT '',
   failText varchar(200) CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT 'Sin Fallas',
   remark varchar(200) CHARACTER SET utf8mb4 COLLATE utf8mb4_unicode_ci NOT NULL DEFAULT 'Sin Observaciones',
   createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   updatedAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
   PRIMARY KEY (id)
)

ENGINE = INNODB,
   AVG_ROW_LENGTH = 910,
   CHARACTER SET utf8mb4,
   COLLATE utf8mb4_general_ci,
   COMMENT = 'Test specimen';

ALTER TABLE specimen
   ADD CONSTRAINT FK_test_specimen_test_sample_id FOREIGN KEY (sample)
   REFERENCES sample (id);
```

Depends On 1



Used By 1

<u>=</u> countSpecimens

insertData

insertSpecimen

selectSpecimen

selectCompareTests

selectTest

== updateSpecimen

data

DeleteTest

7 DeleteTestSample

deleteTest

<u></u> getSpecimenData

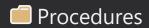
countSpecimens

selectTestNumber

selectTestSpecimen

selectTestsSpecimens

specimenExists



Objects 44

Name	Description
countSpecimens	
deleteTest	
getSpecimenData	
insertData	
<u>insertSample</u>	
insertSpecimen	
selectCompareTests	
<u>selectSample</u>	
selectSampleID	
selectSpecimen	
selectTest	
<u>selectTestData</u>	
<u>selectTests</u>	
<u>updateSpecimen</u>	
selectSampleID selectSpecimen selectTest selectTestData selectTests	



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSample	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE countSpecimens (IN idSample int UNSIGNED)

DETERMINISTIC

BEGIN

SELECT

COUNT(ts.sample) AS counts

FROM specimen ts

WHERE ts.sample = idSample;

END
```

Depends On 2



Used By 1



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	False
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE deleteTest (IN idSpecimen int UNSIGNED)

BEGIN

DELETE LOW_PRIORITY QUICK

FROM specimen ts

WHERE ts.id = idSpecimen;

END
```

Depends On 🛈



Used By

getSpecimenData

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
mySpecimenID	INT		11	

SQL Script

```
PROCEDURE getSpecimenData (IN mySpecimenID int UNSIGNED)
  CREATE TEMPORARY TABLE IF NOT EXISTS myResult AS (SELECT
     td.id,
     td.pressure,
     td.temperature,
     # td.createdAt,
(TIMESTAMPDIFF(MICROSECOND, (SELECT
         td2.createdAt
       FROM data td2
       WHERE td2.specimen = mySpecimenID LIMIT 1), td.createdAt)) AS `difference`
   FROM data td
    WHERE td.specimen = mySpecimenID);
   ts.id AS `idTest`,
        JSON_ARRAYAGG(JSON_OBJECT('id', r.id, 'pressure', r.pressure, 'temperature', r.temperature, 'timekey', r.differe
     FROM myResult r) AS `testData`
 FROM specimen ts
   INNER JOIN sample ts1
     ON ts.sample = ts1.id
 WHERE ts.id = mySpecimenID;
 DROP TEMPORARY TABLE IF EXISTS myResult;
```

Depends On 3



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	False
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	
pressure	DOUBLE	0		
temperature	DOUBLE	0		

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE insertData (IN idSpecimen int UNSIGNED, IN pressure double, IN temperature double)

BEGIN

INSERT INTO data (specimen, pressure, temperature, createdAt)

VALUES (idSpecimen, pressure, temperature, NOW());

END
```

Depends On 3



specimen

Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
standard	VARCHAR	25		
material	VARCHAR	10		
specification	VARCHAR	6		
diamnom	INT		11	
diamreal	INT		11	
wallthick	INT		11	
lenfree	INT		11	
lentotal	INT		11	
targetpressure	INT		11	
targettemp	INT		11	
condPeriod	VARCHAR	15		

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE insertSample (IN standard varchar(25), IN material varchar(10), IN specification varchar(6), IN diamnom int UNSIGNED, IN diamneal int UNSIGNED, IN wallthick int UNSIGNED, IN lenfree int UNSIGNED, IN lentotal int UNSIGNED, IN targetpressure int, IN targettemp int, IN condPeriod varchar(15))

DETERMINISTIC

BEGIN

DECLARE EXIT HANDLER FOR SQLEXCEPTION

BEGIN

GET DIAGNOSTICS CONDITION 1 @sqlstate = RETURNED_SQLSTATE, @message = MESSAGE_TEXT;

SELECT

@message AS `idSample`;

END;
```

```
INSERT INTO sample (standard, condPeriod, material, specification, targettemp, targetpressure, diamreal, diamnom,
wallthick, lenfree, lentotal)

VALUES (standard, condPeriod, material, specification, targettemp, targetpressure, diamreal, diamnom, wallthick
    , lenfree, lentotal);

SELECT
    ts.id AS `idSample`
FROM sample ts
ORDER BY ts.id DESC
LIMIT 1;

END
```

Depends On 2



Used By

insertSpecimen

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	False
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSample	INT		11	
operatorName	VARCHAR	40		
enviromentIn	VARCHAR	20		
testNameIn	VARCHAR	120		
endCapIn	VARCHAR	10		

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE insertSpecimen (IN idSample int UNSIGNED, IN operatorName varchar(40), IN environmentIn varchar(20), IN testNameIn varchar(120), IN endCapIn varchar(10))
BEGIN

INSERT INTO specimen (sample, operator, environment, testName, endCap, createdAt, updatedAt)
VALUES (idSample, operatorName, environmentIn, testNameIn, endCapIn, CURRENT_TIMESTAMP(), CURRENT_TIMESTAMP());

SELECT
ts.id AS `idSpecimen`
FROM specimen ts
ORDER BY ts.id DESC
LIMIT 1;
END
```

Depends On 3



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
arraySpecimens	TEXT			

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectCompareTests (IN arraySpecimens text)

DETERMINISTIC

BEGIN

SELECT

(selectTestSample(sa.id)) AS 'mySample',
 (selectTestSpecimen(se.id)) AS 'mySpecimen',
 (selectTestData(se.id)) AS 'mySpecimen',
 (selectTestData(se.id)) AS 'myData'

FROM specimen se

INNER JOIN sample sa
 ON se.sample = sa.id

WHERE FIND_IN_SET(se.id, arraySpecimens) > 0;
```

Depends On ¹

reportGenerator@%

selectTestSpecimen

selectTestData

specimen

sample

Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSample	INT		11	

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE selectSample (IN idSample int UNSIGNED)
DETERMINISTIC
BEGIN

SELECT

s.id AS 'idSample',
s.standard AS 'standard',
s.material AS 'material',
s.specification AS 'specification',
s.diamreal AS 'diamreal',
s.diamnom AS 'diamnom',
s.wallthick AS 'wallthickness',
s.lenfree AS 'lenfree',
s.lentotal AS 'condPeriod',
s.condPeriod AS 'condPeriod',
s.targettemp AS 'targettemp',
s.targettpressure AS 'targetpressure'
FROM sample s
WHERE s.id = idSample;
```

Depends On ②



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
standard	VARCHAR	25		
material	VARCHAR	10		
specification	VARCHAR	6		
diamnom	INT		11	
diamreal	INT		11	
wallthick	INT		11	
lenfree	INT		11	
lentotal	INT		11	
targetpressure	INT		11	
targettemp	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectSampleID (IN standard varchar(25), IN material varchar(10), IN specification varchar(6), IN diamnom int UNSIGNED, IN diamneal int UNSIGNED, IN wallthick int UNSIGNED, IN lenfree int UNSIGNED, IN lentotal int UNSIGNED, IN targetpressure int, IN targettemp int)

DETERMINISTIC

BEGIN

SELECT

ts.id AS idSample

FROM sample ts

WHERE (ts.standard LIKE standard

AND ts.material LIKE material

AND ts.diamreal e diamreal

AND ts.diamnom = diamnom

AND ts.wallthick = wallthick

AND ts.lenfree = lenfree
```

AND ts.lentotal = lentotal

AND ts.targettemp = targettemp

AND ts.targetpressure = targetpressure);

END

Depends On ②

dataCollector@%

sample

Used By

No items found

selectSpecimen

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE selectSpecimen (IN idSpecimen int UNSIGNED)
DETERMINISTIC
BEGIN

SELECT

s.id AS `idSpecimen`,
s.sample AS `idSample`,
s.enviroment AS `enviroment`,
s.testName AS `testName`,
s.operator AS `operator`,
s.endCap AS `endCap`
FROM specimen s
WHERE s.id = idSpecimen;
```

Depends On 2



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectTest (IN idSpecimen int UNSIGNED)

DETERMINISTIC

BEGIN

SELECT
    (selectTestSample(sa.id)) AS 'mySample',
    (selectTestSpecimen(se.id)) AS 'mySpecimen'

FROM specimen se
    INNER JOIN sample sa
    ON se.sample = sa.id

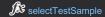
WHERE se.id = idSpecimen;

END
```

Depends On 💿



reportGenerator@%



€ selectTestSpecimen

specimen

sample

Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	

SQL Script

```
PROCEDURE selectTestData (IN idSpecimen int UNSIGNED)
     td.createdAt
   FROM data td
   WHERE td.specimen = idSpecimen
   ORDER BY td.createdAt ASC LIMIT 1);
 WITH groupingData
     td.createdAt,
     AVG(td.pressure) AS 'pressure',
     AVG(td.temperature) AS 'temperature'
   FROM data td
   WHERE td.specimen = idSpecimen
   GROUP BY td.createdAt)
   TIMESTAMPDIFF(SECOND, @initTime, gd.createdAt) AS 'key',
   gd.pressure AS 'pressure',
   gd.temperature AS 'temperature'
  FROM groupingData gd;
```

Depends On 2





selectCompareTests



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectTests ()

DETERMINISTIC

BEGIN

SELECT

ts.id AS idSample,
ts.standard AS standard,
ts.material AS material,
selectTestsSpecimens(ts.id) AS mySpecimens

FROM sample ts;

END
```

Depends On 3

reportGenerator@%



sample

Used By

updateSpecimen

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	False
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	
testName	VARCHAR	150		
operator	VARCHAR	40		
fail	VARCHAR	255		
remark	VARCHAR	255		

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE updateSpecimen (IN idSpecimen int UNSIGNED, IN testName varchar(150), IN operator varchar(40), IN fail varchar(255), IN remark varchar(255))
BEGIN

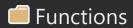
UPDATE LOW_PRIORITY specimen ts
SET ts.testName = testName,
    ts.operator = operator,
    ts.failText = fail,
    ts.remark = remark,
    ts.updatedAt = CURRENT_TIMESTAMP()
WHERE ts.id = idSpecimen;

END
```

Depends On 2



Used By



Objects ³

Name	Description
countSpecimens	
<u>selectTestData</u>	
<u>selectTestLimit</u>	
<u>selectTestNumber</u>	
<u>selectTestSample</u>	
<u>selectTestSpecimen</u>	
<u>selectTestsSpecimens</u>	
<u>specimenExists</u>	



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	
(Result)	INT		11	

SQL Script

Depends On 1



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	
(Result)	JSON	0		

SQL Script

```
FUNCTION selectTestData (idSpecimen int UNSIGNED)
     td.createdAt
   FROM data td
   WHERE td.specimen = idSpecimen
   ORDER BY td.createdAt ASC LIMIT 1);
 WITH groupingData
     td.createdAt,
     AVG(td.pressure) AS 'pressure',
     AVG(td.temperature) AS 'temperature'
   FROM data td
   WHERE td.specimen = idSpecimen
   GROUP BY td.createdAt)
    TIMESTAMPDIFF(SECOND, @initTime, gd.createdAt),
   gd.pressure,
   gd.temperature)) INTO @returnData
  FROM groupingData gd;
```

ΕN

Depends On 🛈



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	
isASC	TINYINT		4	
(Result)	DATETIME		0	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION selectTestLimit (idSpecimen int UNSIGNED, isASC tinyint UNSIGNED)
RETURNS datetime
DETERMINISTIC
BEGIN

SET @Time = NULL;

IF isASC = 0 THEN
SELECT
td.createdAt INTO @Time
FROM data td
WHERE td.specimen = idSpecimen
ORDER BY td.createdAt ASC LIMIT 1;
ELSE
SELECT
td.createdAt INTO @Time
FROM data td
WHERE td.specimen = idSpecimen
ORDER BY td.createdAt ASC LIMIT 1;
ELSE
SELECT
td.createdAt INTO @Time
FROM data td
WHERE td.specimen = idSpecimen
ORDER BY td.createdAt DESC LIMIT 1;
END IF;

RETURN @Time;
END
```

Depends On 1





selectTestSpecimen

selectTestsSpecimens



Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	
(Result)	INT		11	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION selectTestNumber (idSpecimen int UNSIGNED)
RETURNS int(11) UNSIGNED
DETERRINISTIC
BEGIN
SET @testNumber = 0;

SET @sampleID = (SELECT
    ts.sample
    FROM specimen ts
    WHERE ts.id = idSpecimen);

SELECT
    COUNT(*) INTO @testNumber
FROM specimen ts
WHERE ts.id <= idSpecimen
AND ts.sample = @sampleID
GROUP BY ts.sample;

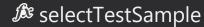
RETURN @testNumber;
END
```

Depends On •



Used By 2





Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSample	INT		11	
(Result)	JSON	0		

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION selectTestSample (idSample int UNSIGNED)
RETURNS json
DETERMINISTIC
BEGIN

SET @result = NULL;

SELECT
JSON_OBJECT('idSample',
sa.id,
'standard',
sa.standard,
'material',
sa.material,
'specification',
sa.specification,
'diameterReal',
sa.diamreal,
'diameterReal',
sa.diamnom,
'wallThickness',
sa.wallthick,
'lengthTotal,
'lengthFree',
sa.lentotal,
'lengthFree',
sa.lentoresure,
'sa.targetTemperature',
sa.targetTemperature',
s
```

sa.condPeriod) INTO @result
FROM sample sa
WHERE sa.id = idSample;

RETURN @result;
END

Depends On
sample

Used By
selectCompareTests
selectTest



Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:05:27
Last Modified	11/6/2024 11:05:27

Parameters

Name	Data Type	Length	Precision	Scale
idSpecimen	INT		11	
(Result)	JSON	0		

SQL Script

```
CREATE

DEFINER = 'root'
FUNCTION selectTestSpecimen (idSpecimen int UNSIGNED)
RETURNS json
DEFERMINISTIC
BEGIN

SET @resuit = NULL;

SET @countSpecimens = countSpecimens(idSpecimen);

SET @beginTime = selectTestLimit(idSpecimen, 0);

SET @endTime = selectTestLimit(idSpecimen, 1);

SET @testNumber = selectTestNumber(idSpecimen, 1);

SELECT

JSON_OBJECT('idSpecimen',
se.id,
'operator',
se.operator,
'environent',
se.environentt,
'beginTime',
DATE_FORMAT(@beginTime,
'Ad/Na/YY %H:%L's'),
'endTime',
DATE_FORMAT(@endTime,
'Ad/Na/YY %H:%L's'),
'duration',
DATE_FORMAT(IMEDIFF(@endTime,
@beginTime),
'Ad/Na/YY %H:%L's'),
'duration',
DATE_FORMAT(IMEDIFF(@endTime,
@beginTime),
'Ad/Na/YY %H:%L's'),
'counts',
```

```
## Security Procession

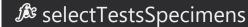
## Security Process

## Se
```

Used By 2

selectCompareTests

selectTest



Description

Properties

Name	Value	
SQL Access	CONTAINS SQL	
Deterministic	True	
Security	Definer	
Definer	root	
SQL Mode	NO_AUTO_VALUE_ON_ZERO	
Language	SQL	
Created	11/6/2024 11:05:27	
Last Modified	11/6/2024 11:05:27	

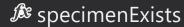
Parameters

Name	Data Type	Length	Precision	Scale
idSample	INT		11	
(Result)	JSON	0		

SQL Script

Depends On 3





Description

Properties

Name	Value	
SQL Access	CONTAINS SQL	
Deterministic	True	
Security	Definer	
Definer	root	
SQL Mode	NO_AUTO_VALUE_ON_ZERO	
Language	SQL	
Created	11/6/2024 11:05:27	
Last Modified	11/6/2024 11:05:27	

Parameters

Name	Data Type	Length	Precision	Scale
idSample	INT		11	
start	VARBINARY	255		
(Result)	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

FUNCTION specimenExists (idSample int UNSIGNED, start varbinary(255))

RETURNS int(11) UNSIGNED

DETERMINISTIC

BEGIN

DECLARE idSpecimen boolean;

SELECT

ts.id INTO idSpecimen

FROM specimen ts

WHERE (ts.sample = idSample

AND ts.start = start)

LIMIT 1;

RETURN idSpecimen;

END
```

Depends On O



Used By

Project / Servers / Local_Development / Databases / data_db / Triggers

Clippers

Objects
Name

DeleteTest

DeleteTestSample

NewSample

DeleteTest

Properties

Name	Value
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Event	DELETE
Timing	BEFORE
Follows	
Precedes	
Created	11/6/2024 11:05:48

SQL Script

```
CREATE

DEFINER = 'root'

TRIGGER DeleteTest

BEFORE DELETE

ON specimen

FOR EACH ROW

BEGIN

DELETE LOW_PRIORITY QUICK

FROM data td

WHERE td.specimen = OLD.id;

END
```

Depends On 3



specimen

Used By

DeleteTestSample

Properties

Name	Value
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Event	DELETE
Timing	AFTER
Follows	
Precedes	
Created	11/6/2024 11:05:48

SQL Script

```
CREATE

DEFINER = 'root'

TRIGGER DeleteTestSample

AFTER DELETE

ON specimen

FOR EACH ROW

BEGIN

SET @countTests = 0;

SELECT

COUNT(*) INTO @countTests

FROM specimen ts

WHERE ts.sample = OLD.sample;

IF @countTests < 1 THEN

DELETE LOW_PRIORITY QUICK

FROM sample

WHERE sample.id = OLD.sample;

END IF;
```

Depends On 2



Used By

NewSample

Properties

Name	Value
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Event	INSERT
Timing	BEFORE
Follows	
Precedes	
Created	11/6/2024 11:05:48

SQL Script

```
TRIGGER NewSample
ON sample
 DECLARE row_id int UNSIGNED;
   s.id INTO row_id
 FROM sample s
 WHERE s.standard LIKE NEW.standard
 AND s.material LIKE NEW.material
 AND s.specification LIKE NEW.specification
 AND s.diamreal = NEW.diamreal
  AND s.diamnom = NEW.diamnom
 AND s.lentotal = NEW.lentotal
 AND s.lenfree = NEW.lenfree
 AND s.wallthick = NEW.wallthick
 AND s.targettemp = new.targettemp
 AND s.targetpressure = NEW.targetpressure;
  IF row_id IS NOT NULL THEN
   SIGNAL SQLSTATE '45000' SET MESSAGE_TEXT = row_id;
```

Depends On **1**



Used By

Hoop Testing



Properties

Name	Value
Collation	utf8mb4_0900_ai_ci

Object Types 4





Procedures



7 Triggers



Objects 10

Description
Conditional Period related to Standard
End Cap related to a Standard
Enviroments related to a Standard
Material of Hoop
Related Materials to a Standard
Test Operator
Settings by Specification
Specification of a Material
Standards: ISO, IRAM,
Test Type by Standard

■ conditional_period

Description

Conditional Period related to Standard

Properties

Name	Value
Engine	InnoDB
Auto Increment	0
Average Row Length	3276
Charset	utf8mb4
Collation	utf8mb4_unicode_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	24/6/2024 22:36:05
Last Modified	1/1/0001 00:00:00

Columns

Key N	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
id □	d	INT		11		True	False	False	True	True		False	False	
st	standard	INT		11		True	False	False	True	False		False	False	
tir	ime	INT		11		True	False	False	True	False	'0'	False	False	
tir	imeType	VARCHAR	3			False	False	False	True	False	'h'	False	False	
ap	aproxTime	INT		11		True	False	False	True	False	'0'	False	False	
aı	aproxType	VARCHAR	3			False	False	False	True	False	'min'	False	False	
m	minwall	INT		11		True	False	False	True	False		False	False	
m	maxwall	INT		11		True	False	False	True	False		False	False	
cr	createdAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
u	updatedtAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
⊞ ⊙		id	True	None	0
	FK_conditional_period_standard_id	standard	False	None	0

Foreign Keys

Name	Columns	Delete Rule	Update Rule
FK_conditional_period_standard_id	id	N/S	N/S

SQL Script

```
CREATE TABLE conditional_period (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   standard int UNSIGNED NOT NULL,
   time int UNSIGNED NOT NULL DEFAULT 0,
   timeType varchar(3) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL DEFAULT 'h',
   aproxTime int UNSIGNED NOT NULL DEFAULT 0,
   aproxType varchar(3) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL DEFAULT 'min',
   minwall int UNSIGNED NOT NULL,
   maxwall int UNSIGNED NOT NULL,
   createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   updatedtAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
   PRIMARY KEY (id)
)

ENGINE = INNODB,
   AVG_ROW_LENGTH = 3276,
   CHARACTER SET utf8mb4,
   COLLATE utf8mb4_unicode_ci,
   COMMENT = 'Conditional Period related to Standard';

ALTER TABLE conditional_period
   ADD CONSTRAINT FK_conditional_period_standard_id FOREIGN KEY (standard)
   REFERENCES standard (id);
```

Depends On 🕡



Used By 🕖

selectConditionalPeriods

deleteConditionalPeriod

insertConditionalPeriod

selectStandarsComplete

sexistConditionalPeriod

selectConditionalPeriods_has_Standard

5 deleteStandardChilds

endcap

Description

End Cap related to a Standard

Properties

Name	Value
Engine	InnoDB
Auto Increment	0
Average Row Length	5461
Charset	latin1
Collation	latin1_swedish_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	26/6/2024 20:21:13
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary		Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	standard	INT		11		True	False	False	True	False		False	False	
	name	VARCHAR	15			False	False	False	True	False		False	False	
	createdAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
	updatedAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
		id	True	None	0
	FK_endcap_standard_id	standard	False	None	0

Foreign Keys

Name	Columns	Delete Rule	Update Rule
FK_endcap_standard_id	id	N/S	N/S

SQL Script

```
CREATE TABLE endcap (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   standard int UNSIGNED NOT NULL,
   name varchar(15) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
   createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   updatedAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   PRIMARY KEY (id)
  )
   ENGINE = INNODB,
   AVG_ROW_LENGTH = 5461,
   CHARACTER SET latin1,
   COLLATE latin1_swedish_ci,
   COMMENT = 'End Cap related to a Standard';

ALTER TABLE endcap
   ADD CONSTRAINT FK_endcap_standard_id FOREIGN KEY (standard)
   REFERENCES standard (id) ON DELETE CASCADE ON UPDATE CASCADE;
```

Depends On 1



Used By 🕡

selectEndCaps

deleteEndCap

insertEndCap

selectStandarsComplete

existEndCap

selectEndCaps_has_Standard

€ deleteStandardChilds

u enviroment

Description

Enviroments related to a Standard

Properties

Name	Value
Engine	InnoDB
Auto Increment	0
Average Row Length	4096
Charset	latin1
Collation	latin1_swedish_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	26/6/2024 20:38:44
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	standard	INT		11		True	False	False	True	False		False	False	
	inside	VARCHAR	30			False	False	False	True	False		False	False	
	outside	VARCHAR	30			False	False	False	False	False	'No Fluid'	False	False	
	createdAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
	updatedAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
₩		id	True	None	0
	FK_enviroment_standard_id	standard	False	None	0

Foreign Keys

Name	Columns	Delete Rule	Update Rule
FK_enviroment_standard_id	id	N/S	N/S

SQL Script

```
CREATE TABLE enviroment (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   standard int UNSIGNED NOT NULL,
   inside varchar(30) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
   outside varchar(30) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci DEFAULT 'No Fluid',
   createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   updatedAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
   PRIMARY KEY (id)
)
ENGINE = INNODB,
AVG_ROW_LENGTH = 4096,
CHARACTER SET latin1,
COLLATE latin1_swedish_ci,
COMMENT = 'Enviroments related to a Standard';

ALTER TABLE enviroment
ADD CONSTRAINT FK_enviroment_standard_id FOREIGN KEY (standard)
REFERENCES standard (id) ON DELETE CASCADE ON UPDATE CASCADE;
```

Depends On 1



Used By 🕡

selectEnviroment

deleteEnviroment

insertEnviroment

selectStandarsComplete

sexistEnviroment

selectEnviroments_has_Standard

5 deleteStandardChilds



Description

Material of Hoop

Properties

Name	Value
Engine	InnoDB
Auto Increment	14
Average Row Length	16384
Charset	utf8mb4
Collation	utf8mb4_general_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	25/6/2024 00:48:45
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary		Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	name	VARCHAR	15			False	False	False	True	False		False	False	
	description	VARCHAR	120			False	False	False	True	False	'Sin descripción'	False	False	
	createdAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
	updatedAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
⊞ ⊙		id	True	None	0

Triggers

Mana	T	Format.	0	D. C
Name	Type	Event	Security	Definer

deleteMaterialChilds BEFORE DELETE Definer root

Permissions

Action	Owner
SELECT	databaseManager@%

SQL Script

```
CREATE TABLE material (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   name varchar(15) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
   description varchar(120) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL DEFAULT 'Sin descripción',
   createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   updatedAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
   PRIMARY KEY (id)
)
ENGINE = INNODB,
AVG_ROW_LENGTH = 16384,
CHARACTER SET utf8mb4,
COLLATE utf8mb4_general_ci,
COMMENT = 'Material of Hoop';
```

Depends On 1



Used By 🚯

selectMaterials

deleteMaterial

insertMaterial

insertRelatedMaterial

insertSpecification

selectMaterialsJSON

updateMaterial

deleteMaterialChilds

material_has_standard

specification

selectStandarsComplete

🏂 existMaterial

selectMaterials_has_Standard

material_has_standard

Description

Related Materials to a Standard

Properties

Name	Value
Engine	InnoDB
Auto Increment	0
Average Row Length	8192
Charset	utf8mb4
Collation	utf8mb4_0900_ai_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	26/6/2024 20:20:08
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	material	INT		11		True	False	False	True	False		False	False	
	standard	INT		11		True	False	False	True	False		False	False	
	type	VARCHAR	25			False	False	False	True	False	'Plastico'	False	False	
	createdAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
	updatedAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
⊞ ⊙		id	True	None	0

FK_material_has_standard_material_id	material	False	None	0
FK_material_has_standard_standard_id	standard	False	None	0

Foreign Keys

Name	Columns	Delete Rule	Update Rule
FK_material_has_standard_material_id	id	N/S	N/S
FK_material_has_standard_id	id	N/S	N/S

SQL Script

```
CREATE TABLE material_has_standard (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   material int UNSIGNED NOT NULL,
   standard int UNSIGNED NOT NULL,
   standard int UNSIGNED NOT NULL,
   type varchar(25) NOT NULL DEFAULT 'Plastico',
   createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   updatedAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   primary KEY (id)
)

ENGINE = INNODB,
   AVG_ROW_LENGTH = 8192,
   CHARACTER SET utf8mb4,
   COLLATE utf8mb4_0900_ai_ci,
   COMMENT = 'Related Materials to a Standard';

ALTER TABLE material_has_standard
   ADD CONSTRAINT FK_material_has_standard_material_id FOREIGN KEY (material)
   REFERENCES material (id) ON DELETE CASCADE ON UPDATE CASCADE;

ALTER TABLE material_has_standard
   ADD CONSTRAINT FK_material_has_standard_standard_id FOREIGN KEY (standard)
   REFERENCES standard (id) ON DELETE CASCADE ON UPDATE CASCADE;
```

Depends On 2



standard

Used By 3

selectMaterials

deleteRelatedMaterial

insertRelatedMaterial

selectStandarsComplete

existRelatedMaterial

selectMaterials_has_Standard

5 deleteMaterialChilds

deleteStandardChilds

u operator

Description

Test Operator

Properties

Name	Value
Engine	InnoDB
Auto Increment	0
Average Row Length	2730
Charset	utf8mb4
Collation	utf8mb4_0900_ai_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	26/6/2024 20:20:18
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary		Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	dni	INT		11		True	False	False	True	False		False	False	
	name	VARCHAR	50			False	False	False	True	False		False	False	
	familyname	VARCHAR	50			False	False	False	True	False		False	False	
	createdAt	DATETIME		4		False	False	False	True	False	CURRENT_TIMESTAMP(4)	False	False	
	updatedAt	DATETIME		4		False	False	False	True	False	CURRENT_TIMESTAMP(4)	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
⊞		id	True	None	0

SQL Script

```
CREATE TABLE operator (
id int UNSIGNED NOT NULL AUTO_INCREMENT,
```

```
dni int UNSIGNED NOT NULL,
   name varchar(50) NOT NULL,
   familyname varchar(50) NOT NULL,
  createdAt datetime(4) NOT NULL DEFAULT CURRENT_TIMESTAMP(4),
updatedAt datetime(4) NOT NULL DEFAULT CURRENT_TIMESTAMP(4) ON UPDATE CURRENT_TIMESTAMP(4),
ENGINE = INNODB,

AVG_ROW_LENGTH = 2730,

CHARACTER SET utf8mb4,
COLLATE utf8mb4_0900_ai_ci,
```

Depends On

Used By











Author: Ezequiel Augusto Stanganelli

Created: 27/06/2024

settings_specification

Description

Settings by Specification

Properties

Name	Value
Engine	InnoDB
Auto Increment	0
Average Row Length	8192
Charset	utf8mb4
Collation	utf8mb4_general_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	27/6/2024 10:56:28
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	specification	INT		11		True	False	False	True	False		False	False	
	time	SMALLINT		6		True	False	False	True	False		False	False	
	timeType	VARCHAR	3			False	False	False	True	False	'h'	False	False	
	temperature	TINYINT		4		True	False	False	True	False		False	False	
	createdAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
	updatedAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
⊞ ⊙		id	True	None	0

Foreign Keys

Name	Columns	Delete Rule	Update Rule
FK_specification_configuration_specification_id	id	N/S	N/S

SQL Script

```
CREATE TABLE settings_specification (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   specification int UNSIGNED NOT NULL,
   time smallint UNSIGNED NOT NULL,
   timeType varchar(3) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL DEFAULT 'h',
   temperature tinyint UNSIGNED NOT NULL,
   createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   updatedAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
   PRIMARY KEY (id)
)

ENGINE = INNODB,
   AVG_ROW_LENGTH = 8192,
   CHARACTER SET utf8mb4,
   COLLATE utf8mb4_general_ci,
   COMMENT = 'Settings by Specification';

ALTER TABLE settings_specification
   ADD INDEX FK_period_by_material_material_id (specification);

ALTER TABLE settings_specification
   ADD CONSTRAINT FK_specification_configuration_specification_id FOREIGN KEY (specification)
   REFERENCES specification (id) ON UPDATE CASCADE;
```

Depends On 1



Used By 🕡

selectSettingsSpecification

deleteSettingsSpecification

insertSettingsSpecification

updateSettingsSpecification

sexistSettingsSpecification

selectSettingsSpecification

deleteConfigurationsChilds

specification

Description

Specification of a Material

Properties

Name	Value
Engine	InnoDB
Auto Increment	22
Average Row Length	16384
Charset	utf8mb4
Collation	utf8mb4_general_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	26/6/2024 20:20:26
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	material	INT		11		True	False	False	True	False		False	False	
	name	VARCHAR	15			False	False	False	True	False		False	False	
	description	VARCHAR	120			False	False	False	True	False	'Sin Descripción'	False	False	
	createdAt	DATETIME		0		False	False	False	True	False		False	False	
	updatedAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
⊞		id	True	None	0
	FK_specification_material_id	material	False	None	0

Triggers

Name	Туре	Event	Security	Definer
deleteConfigurationsChilds	BEFORE	DELETE	Definer	root

Foreign Keys

Name	Columns	Delete Rule	Update Rule
FK_specification_material_id	id	N/S	N/S

SQL Script

```
CREATE TABLE specification (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   material int UNSIGNED NOT NULL,
   name varchar(15) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
   description varchar(120) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL DEFAULT 'Sin Descripción',
   createdAt datetime NOT NULL,
   updatedAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
   PRIMARY KEY (id)
)

ENGINE = INNODB,
   AVG_ROW_LENGTH = 16384,
   CHARACTER SET utf8mb4,
   COLLATE utf8mb4_general_ci,
   COMMENT = 'Specification of a Material';

ALTER TABLE specification
   ADD CONSTRAINT FK_specification_material_id FOREIGN KEY (material)
   REFERENCES material (id) ON UPDATE CASCADE;
```

Depends On 1



Used By 12

- selectSpecifications
- deleteSpecification
- insertSettingsSpecification
- insertSpecification
- updateSpecification
- settings_specification
- deleteConfigurationsChilds
- sexistConfiguration
- sexistSpecification
- 🏂 existSpecimen
- selectSpecifications
- **5** deleteMaterialChilds

standard

Description

Standards: ISO, IRAM, ...

Properties

Name	Value
Engine	InnoDB
Auto Increment	8
Average Row Length	16384
Charset	utf8mb4
Collation	utf8mb4_general_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	11/6/2024 11:06:02
Last Modified	26/6/2024 20:18:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
₩ ##	id	INT		11		True	False	False	True	True		False	False	
	name	VARCHAR	60			False	False	False	True	False		False	False	
	createdAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
	updatedAt	DATETIME		0		False	False	False	False	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
⊞		id	True	None	0

Triggers

Name	Туре	Event	Security	Definer
deleteStandardChilds	BEFORE	DELETE	Definer	root

SQL Script

```
CREATE TABLE standard (
  name varchar(60) CHARACTER SET utf8mb4 COLLATE utf8mb4_0900_ai_ci NOT NULL,
  createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
updatedAt datetime DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
CHARACTER SET utf8mb4,
COLLATE utf8mb4_general_ci,
```

Depends On

Used By 19

selectMaterials

selectStandards deleteStandard

insertConditionalPeriod

insertEndCap

insertEnviroment

insertRelatedMaterial

insertStandard

insertTestType

selectStandardsJSON

updateStandard

conditional_period

endcap

enviroment

material_has_standard

5 deleteStandardChilds

test_type

selectStandarsComplete

sexistStandard

test_type

Description

Test Type by Standard

Properties

Name	Value
Engine	InnoDB
Auto Increment	0
Average Row Length	0
Charset	utf8mb4
Collation	utf8mb4_0900_ai_ci
Row Format	Dynamic
Min Rows	0
Max Rows	0
Checksum	False
Page Checksum	True
Pack Keys	False
Delay Key Write	False
Is Partitioned	False
Encryption	False
Persistent Statistics	DEFAULT
Auto Recalculate Statistics	DEFAULT
Sample Pages	0
Created	26/6/2024 20:21:00
Last Modified	1/1/0001 00:00:00

Columns

Key	Name	Data Type	Length	Precision	Scale	Unsigned	Zerofill	Binary	Not Null	Auto Increment	Default	Virtual	Invisible	Description
	id	INT		11		True	False	False	True	True		False	False	
	standard	INT		11		True	False	False	True	False		False	False	
	testType	VARCHAR	150			False	False	False	True	False		False	False	
	createdAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	
	updatedAt	DATETIME		0		False	False	False	True	False	CURRENT_TIMESTAMP	False	False	

Indexes

Key	Name	Columns	Unique	Туре	Key Lengths
ಪಾ		id	True	None	0
	FK_test_type_standard_id	standard	False	None	0

Foreign Keys

Name	Columns	Delete Rule	Update Rule
FK_test_type_standard_id	id	N/S	N/S

SQL Script

```
CREATE TABLE test_type (
   id int UNSIGNED NOT NULL AUTO_INCREMENT,
   standard int UNSIGNED NOT NULL,
   testType varchar(150) NOT NULL,
   createdAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP,
   updatedAt datetime NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,
   PRIMARY KEY (id)
)
ENGINE = INNODB,
CHARACTER SET utf8mb4,
COLLATE utf8mb4_0900_ai_ci,
COMMENT = 'Test Type by Standard';

ALTER TABLE test_type
ADD CONSTRAINT FK_test_type_standard_id FOREIGN KEY (standard)
REFERENCES standard (id) ON DELETE CASCADE ON UPDATE CASCADE;
```

Depends On **①**



Used By 6

selectTestType

deleteTestType

insertTestType

selectTest_Type

deleteStandardChilds

Procedures

Objects 38

Name	Description
deleteConditionalPeriod	
deleteEndCap	
deleteEnviroment	
deleteMaterial	
deleteOperator	
deleteRelatedMaterial	
deleteSettingsSpecification	
deleteSpecification	
deleteStandard	
deleteTestType	
insertConditionalPeriod	Insertion of new Conditional Period
insertEndCap	
insertEnviroment	
insertMaterial	
insertOperator	
insertRelatedMaterial	
insertSettingsSpecification	
insertSpecification	
insertStandard	
<u>insertTestType</u>	
selectConditionalPeriods	Selection of Conditional Periods by Standard
selectEndCaps	
selectEnviroment	
selectMaterials	
<u>selectMaterialsJSON</u>	
<u>selectOperators</u>	
selectOperatorsJSON	
<u>selectSettingsSpecification</u>	
selectSpecifications	
selectStandards	
selectStandardsJSON	
<u>selectStandarsComplete</u>	
<u>selectTestType</u>	

<u>updateMaterial</u>	
<u>updateOperator</u>	
<u>updateSettingsSpecification</u>	
updateSpecification	
updateStandard	

deleteConditionalPeriod

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	24/6/2024 22:49:34
Last Modified	24/6/2024 22:49:34

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE deleteConditionalPeriod (IN id int UNSIGNED)

DETERMINISTIC

BEGIN

If (existConditionalPeriod(id) = TRUE) THEN

DELETE LOW_PRIORITY QUICK

FROM conditional_period cp

WHERE cp.id = id

LIMIT 1;

SELECT

'Successful Deleted!' AS `response`;

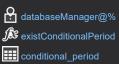
ELSE

SELECT

'Unsuccessful Deleted!' AS `response`;

END IF;
```

Depends On 3



Used By

■ deleteEndCap

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE deleteEndCap (IN id int UNSIGNED)

DETERMINISTIC

BEGIN

IF (existEndCap(id) = TRUE) THEN

DELETE LOW_PRIORITY QUICK

FROM endcap cp

WHERE cp.id = id

LIMIT 1;

SELECT

'Successful Deleted!' AS `response`;

ELSE

SELECT

'Unsuccessful Deleted!' AS `response`;

END IF;
```

Depends On 3

databaseManager@%



endcap

Used By

deleteEnviroment

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE deleteEnviroment (IN id int UNSIGNED)

DETERMINISTIC

BEGIN

IF (existEnviroment(id) = TRUE) THEN

DELETE LOW_PRIORITY QUICK

FROM enviroment e

WHERE e.id = id

LIMIT 1;

SELECT

'Successful Deleted!' AS `response`;

ELSE

SELECT

'Unsuccessful Deleted!' AS `response`;

END IF;
```

Depends On 3

databaseManager@%

enviroment enviroment

Used By

deleteMaterial

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:01
Last Modified	11/6/2024 11:06:01

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE deleteMaterial (IN id int UNSIGNED)

DETERMINISTIC

BEGIN

IF (existMaterial(id) = TRUE) THEN

DELETE LOW_PRIORITY QUICK

FROM material m

WHERE m.id = id

LIMIT 1;

SELECT

'Successful Deleted!' AS `response`;

ELSE

SELECT

'Unsuccessful Deleted!' AS `response`;

END IF;
```

Depends On 3







Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:03
Last Modified	11/6/2024 11:06:03

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE deleteOperator (IN id int UNSIGNED)

DETERMINISTIC

BEGIN

IF (existOperator(id) = TRUE) THEN

DELETE LOW_PRIORITY QUICK

FROM operator o

WHERE o.id = id

LIMIT 1;

SELECT

'Successful Deleted!' AS `response`;

ELSE

SELECT

'Unsuccessful Deleted!' AS `response`;

END IF;
```

Depends On 3

databaseManager@%



Used By

deleteRelatedMaterial

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE deleteRelatedMaterial (IN id int UNSIGNED)
DETERMINISTIC
BEGIN

IF (existRelatedMaterial(id) = TRUE) THEN
DELETE LOW_PRIORITY QUICK
FROM material_has_standard mhs
WHERE mhs.id = id
LIMIT 1;

SELECT
'Successful Deleted!' AS `response`;
ELSE
SELECT
'Unsuccessful Deleted!' AS `response`;
END IF;

END
```

Depends On 3

databaseManager@%

existRelatedMaterial
material_has_standard

Used By

deleteSettingsSpecification

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
Language	sqL
Created	27/6/2024 11:04:58
Last Modified	27/6/2024 11:04:58

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE deleteSettingsSpecification (IN id int UNSIGNED)

DETERMINISTIC

BEGIN

IF (existSettingsSpecification(id) = TRUE) THEN

DELETE LOW_PRIORITY QUICK

FROM settings_specification sc

WHERE sc.id = id

LIMIT 1;

SELECT

'Successful Deleted!' AS `response`;

ELSE

SELECT

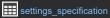
'Unsuccessful Deleted!' AS `response`;

END IF;
```

Depends On







Used By

deleteSpecification

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:01
Last Modified	11/6/2024 11:06:01

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE deleteSpecification (IN id int UNSIGNED)
DETERMINISTIC
BEGIN

IF (existSpecification(id) = TRUE) THEN
DELETE LOW_PRIORITY QUICK
FROM specification s
WHERE s.id = id
LIMIT 1;

SELECT
'Successful Deleted!' AS `response`;
ELSE
SELECT
'Unsuccessful Deleted!' AS `response`;
END IF;

END
```

Depends On 3

databaseManager@%



Used By

deleteStandard

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE deleteStandard (IN id int UNSIGNED)
DETERMINISTIC
BEGIN

IF (existStandard(id) = TRUE) THEN
DELETE LOW_PRIORITY QUICK
FROM standard s
WHERE s.id = id
LIMIT 1;

SELECT
'Successful Deleted!' AS `response`;
ELSE
SELECT
'Unsuccessful Deleted!' AS `response`;
END IF;

END
```

Depends On 3

databaseManager@%



Used By

deleteTestType

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE deleteTestType (IN id int UNSIGNED)

DETERMINISTIC

BEGIN

IF (existTest_Type(id) = TRUE) THEN

DELETE LOW_PRIORITY QUICK

FROM test_type tt

WHERE tt.id = id

LIMIT 1;

SELECT

'Successful Deleted!' AS `response`;

ELSE

SELECT

'Unsuccessful Deleted!' AS `response`;

END IF;
```

Depends On 3

databaseManager@%

continuous databaseManager@%

continuous databaseManager@%

continuous databaseManager@%

continuous databaseManager@%

continuous databaseManager@%

Used By

insertConditionalPeriod

Description

Insertion of new Conditional Period

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	24/6/2024 22:41:34
Last Modified	24/6/2024 22:41:34

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
minwall	INT		11	
maxwall	INT		11	
time	INT		11	
timeType	VARCHAR	3		
aproxTime	INT		11	
aproxType	VARCHAR	3		

```
CREATE

DEFINER = 'root'

PROCEDURE insertConditionalPeriod (IN idStandard int UNSIGNED, IN minwall int UNSIGNED, IN maxwall int UNSIGNED, IN time int UNSIGNED, IN timeType varchar(3), IN aproxTime int UNSIGNED, IN aproxType varchar(3))

DETERMINISTIC

COMMENT 'Insertion of new Conditional Period'

BEGIN

DECLARE elements int UNSIGNED;

SELECT

COUNT(*) INTO elements

FROM conditional_period cp

WHERE cp.standard = idStandard

AND cp.maxwall = minwall

AND cp.maxwall = maxwall

AND cp.time = time

AND cp.timeType LIKE timeType

AND cp.aproxTime = aproxTime

AND cp.aproxType LIKE aproxType;

IF elements = 0 THEN

INSERT HIGH_PRIORITY INTO conditional_period (standard, time, timeType, aproxTime, aproxType, minwall, maxwall)
```

```
VALUES (idStandard, time, timeType, aproxTime, aproxType, minwall, maxwall);

SELECT
cp.id AS 'key',
CONCAT(CONVERT(cp.time, char), ' ', cp.timeType, ' ± ', CONVERT(cp.aproxTime, char), '
', cp.aproxType) AS 'condperiod',
cp.minwall AS 'minwall',
cp.maxwall AS 'maxwall'
FROM conditional_period cp
WHERE cp.standard = idStandard
AND cp.minwall = minwall
AND cp.maxwall = maxwall
AND cp.time LIKE time;

ELSE
SELECT
'Already Exist!' AS 'response';
END IF;

END
```

Depends On 3

databaseManager@%
conditional_period

standard

Used By

■ insertEndCap

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
endCap	VARBINARY	15		

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE insertEndCap (IN idStandard int UNSIGNED, IN endCap varbinary(15))
DETERMINISTIC
BEGIN

DECLARE elements int UNSIGNED;

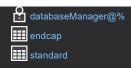
SELECT
COUNT(*) INTO elements
FROM endCap ec
WHERE ec.name = endCap
AND ec.name = endCap;

If elements = 0 THEN
INSERT HIGH_PRIORITY INTO endCap (standard, name)
VALUES (idStandard, endCap);

SELECT
ec.id AS 'key',
ec.name AS 'endCap'
FROM endCap ec
WHERE ec.name = endCap
AND ec.name = endCap
AND ec.name = endCap;

ELSE
SELECT
'Already Exist!' AS 'response';
END IF;
```

Depends On 3



Used By

insertEnviroment

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	26/6/2024 20:38:57
Last Modified	26/6/2024 20:38:57

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
insideFluid	VARCHAR	30		

```
CREATE

DEFINER = 'root'

PROCEDURE insertEnviroment (IN idStandard int UNSIGNED, IN insideFluid varchar(30))

DETERMINISTIC

BEGIN

DECLARE elements int UNSIGNED;

SELECT

COUNT(*) INTO elements

FROM enviroment e

WHERE e.standard = idStandard

AND e.inside LIKE insideFluid;

IF elements = 0 THEN

INSERT HIGH_PROBITY INTO enviroment (standard, inside)

VALUES (idStandard, insideFluid);

SELECT

e.id AS `key',
e.inside AS `insideFluid'

f e.outside AS `outsideFluid'

FROM enviroment e

WHERE e.standard = idStandard

AND e.inside LIKE insideFluid'

FROM enviroment e

WHERE e.standard = idStandard

AND e.inside LIKE insideFluid;

ELSE

SELECT

'Already Exist!' AS `response';

END IF;
```

Depends On databaseManager@%

enviroment

standard

Used By

insertMaterial

Description

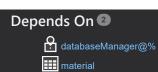
Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	25/6/2024 00:54:27
Last Modified	25/6/2024 00:54:27

Parameters

Name	Data Type	Length	Precision	Scale
namelN	VARCHAR	20		
descriptionIN	VARCHAR	60		

```
PROCEDURE insertMaterial (IN nameIN varchar(20), IN descriptionIN varchar(60))
 DECLARE selected bigint;
 DECLARE result json;
   COUNT(*) INTO selected
 FROM material m
 WHERE m.name LIKE nameIN;
  IF (selected = 0) THEN
   INSERT INTO material (name, description)
     VALUES (nameIN, descriptionIN);
     m.id AS `key`,
     m.name AS `material`,
     m.description AS `description`,
     '[]' AS `specifications`
   FROM material m
    WHERE m.name = nameIN;
      'Already Exists!' AS `response`;
```



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:03
Last Modified	11/6/2024 11:06:03

Parameters

Name	Data Type	Length	Precision	Scale
dni	INT		11	
name	VARCHAR	20		
familyname	VARCHAR	30		

```
CREATE

DEFINER = 'root'

PROCEDURE insertOperator (IN dni int UNSIGNED, IN name varchar(20), IN familyname varchar(30))

DETERMINISTIC

BEGIN

DECLARE selected bigint;

SELECT

COUNT(') INTO selected

FROM operator o

WHERE o.dni = dni;

If (selected = 0) THEN

INSERT INTO operator (dni, name, familyname)

VALUES (dni, name, familyname);

SELECT

o.id AS 'key',
o.dni AS 'dni',
o.name AS 'name',
o.familyname AS 'familyName'

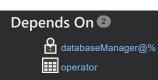
FROM operator o

WHERE o.dni = dni;

ELSE

SELECT

'Already Exists!' AS 'response';
END IF;
```



Used By

■ insertRelatedMaterial

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	False
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
idMaterial	INT		11	

```
PROCEDURE insertRelatedMaterial (IN idStandard int UNSIGNED, IN idMaterial int UNSIGNED)
 DECLARE elements int UNSIGNED;
   COUNT(*) INTO elements
  FROM material_has_standard mhs
 WHERE mhs.material = idMaterial
 AND mhs.standard = idStandard;
  IF elements = 0 THEN
    INSERT HIGH_PRIORITY INTO material_has_standard (material, standard)
     VALUES (idMaterial, idStandard);
     mhs.id AS `key`,
     m.id AS `idMaterial`,
     m.name AS `material`
   FROM material_has_standard mhs
     INNER JOIN material m
       ON mhs.material = m.id
    WHERE mhs.material = idMaterial
   AND mhs.standard = idStandard;
      'Already Exist!' AS `response`;
```

Depends On databaseManager@% material_has_standard material standard

Used By

insertSettingsSpecification

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
Language	sqL
Created	27/6/2024 11:01:28
Last Modified	27/6/2024 11:01:28

Parameters

Name	Data Type	Length	Precision	Scale
specificationIN	INT		11	
timeIN	SMALLINT		6	
timeTypeIN	VARCHAR	1		
temperatureIN	TINYINT		4	

```
PROCEDURE insertSettingsSpecification (IN specificationIN int UNSIGNED, IN timeIN smallint, IN timeTypeIN varchar(1
), IN temperatureIN tinyint)
 DECLARE selected bigint;
   COUNT(*) INTO selected
  FROM settings_specification sc
 WHERE sc.specification = specificationIN
 AND sc.time = timeIN
  AND sc.temperature = temperatureIN;
  IF (selected = 0) THEN
   INSERT INTO settings_specification (specification, time, timeType, temperature)
     VALUES (specificationIN, timeIN, timeTypeIN, temperatureIN);
     sc.id AS `key`,
     sc.time AS `time`,
     sc.timeType AS `timeType`,
     sc.temperature AS `temperature`
    FROM settings_specification sc
    WHERE sc.specification = specificationIN
    AND sc.time = timeIN
    AND sc.temperature = temperatureIN;
      'Already Exists!' AS `response`;
```

insertSpecification

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	25/6/2024 00:54:16
Last Modified	25/6/2024 00:54:16

Parameters

Name	Data Type	Length	Precision	Scale
materialIN	INT		11	
namelN	VARCHAR	20		
descriptionIN	VARCHAR	60		

```
PROCEDURE insertSpecification (IN materialIN int UNSIGNED, IN nameIN varchar(20), IN descriptionIN varchar(60))
 DECLARE selected bigint;
   COUNT(*) INTO selected
 FROM specification s
 WHERE s.name = nameIN
 AND s.material = materialIN;
 IF (selected = 0) THEN
   INSERT INTO specification (material, name, description)
     VALUES (materialIN, nameIN, descriptionIN);
     s.id AS `key`,
     s.name AS `specification`,
     s.description AS `description`,
      '[]' AS `configurations`
   FROM specification s
   WHERE s.name LIKE nameIN;
      'Already Exists!' AS `response`;
```

Depends On databaseManager@%

specification

material

Used By



Properties

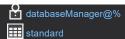
Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
nameln	VARCHAR	40		

SQL Script

Depends On 2



Used By

insertTestType

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	13/6/2024 01:29:06
Last Modified	13/6/2024 01:29:06

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
testTypeIN	VARBINARY	255		

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE insertTestType (IN idStandard int UNSIGNED, IN testTypeIN varbinary(255))
DETERMINISTIC
BEGIN

DECLARE elements int UNSIGNED;

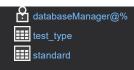
SELECT
COUNT(*) INTO elements
FROM test_type tt
WHERE tt.standard = idStandard
AND tt.testType = testTypeIN;

If elements = 0 THEN
INSERT HIGH_PRIORITY INTO test_type (standard, testType)
VALUES (idStandard, testTypeIN);

SELECT
tt.id AS `key`,
tt.testType AS `testtype`
FROM test_type tt
WHERE tt.standard = idStandard
AND tt.testType = testTypeIN;

ELSE
SELECT
'Already Exist!' AS `response`;
END IF;
```

Depends On 3



Used By



selectConditionalPeriods

Description

Selection of Conditional Periods by Standard

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	24/6/2024 20:39:58
Last Modified	24/6/2024 20:39:58

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	

SQL Script

```
{\tt PROCEDURE} \ \ {\tt selectConditionalPeriods} \ \ ({\tt IN} \ \ {\tt idStandard} \ \ {\tt int} \ \ {\tt UNSIGNED})
   cp.id AS idCondPeriod,
    cp.minwall AS minWall,
    cp.maxwall AS maxWall,
   (CONCAT(CONVERT(cp.time, char), ' ', cp.timeType, ' ± ', CONVERT(cp.aproxTime, char), '
', cp.aproxType)) AS condPeriod
 FROM conditional_period cp
  WHERE cp.standard = idStandard
  ORDER BY cp.minwall;
```

Depends On 2



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	False
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectEndCaps (IN idStandard int UNSIGNED)

BEGIN

SELECT

e.id AS `id`,

e.name AS `endCap`

FROM endcap e

WHERE e.standard = idStandard;
```

Depends On 2



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	26/6/2024 19:08:54
Last Modified	26/6/2024 19:08:54

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	

SQL Script

```
CREATE
DEFINER = 'root'
PROCEDURE selectEnviroment (IN idStandard int UNSIGNED)
DETERMINISTIC
BEGIN

SELECT
ehs.id,
ehs.inside#,
#ehs.outside
FROM enviroment ehs
WHERE ehs.standard = idStandard;

END
```

Depends On 2



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	

SQL Script

```
PROCEDURE selectMaterials (IN idStandard int UNSIGNED)
   mhs.material AS idMaterial,
   m.name AS materialName,
  m.description AS materialDescription,
   mhs.type AS materialType
 FROM material m
   INNER JOIN material_has_standard mhs
     ON m.id = mhs.material
 WHERE mhs.standard = standard;
```

Depends On 4

dataCollector@%



material_has_standard



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectMaterialsJSON ()

DETERMINISTIC

BEGIN

SELECT

JSON_ARRAYAGG(JSON_OBJECT('key', m.id,
    'material', m.name,
    'description', (IF(m.description IS NULL, 'Sin Descripción', m.description)),
    'specifications', (selectSpecifications(m.id)))) AS `materials`

FROM material m;

END
```

Depends On 3

databaseManager@%
selectSpecifications

material

Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:03
Last Modified	11/6/2024 11:06:03

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectOperators ()

DETERMINISTIC

BEGIN

SELECT

o.id,
o.dni,
o.name,
o.familyname

FROM operator o;
```

Depends On 2



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:03
Last Modified	11/6/2024 11:06:03

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectOperatorsJSON ()

DETERMINISTIC

BEGIN

SELECT

JSON_ARRAYAGG(JSON_OBJECT('key', o.id,
 'dni', o.dni,
 'name', o.name,
 'familyName', o.familyname)) AS `operators`

FROM operator o;

END
```

Depends On 2



Used By

selectSettingsSpecification

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
Language	sqL
Created	27/6/2024 11:01:28
Last Modified	27/6/2024 11:01:28

Parameters

Name	Data Type	Length	Precision	Scale
idSpecification	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectSettingsSpecification (IN idSpecification int UNSIGNED)

DETERMINISTIC

BEGIN

SELECT

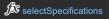
sc.id AS idSetting,
sc.time AS time,
sc.time AS time,
sc.timeType AS timeType,
sc.temperature AS temperature

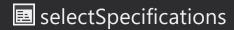
FROM settings_specification sc
WHERE sc.specification = idSpecification;

END
```

Depends On 2







Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:01
Last Modified	11/6/2024 11:06:01

Parameters

Name	Data Type	Length	Precision	Scale
idMaterial	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectSpecifications (IN idMaterial int UNSIGNED)

DETERMINISTIC

BEGIN

SELECT

s.id AS idSpecification,
s.name AS nameSpecification,
s.description AS descriptionSpecification

FROM specification s

WHERE s.material = idMaterial;
```

Depends On 2







Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectStandards ()

DETERMINISTIC

BEGIN

SELECT

s.id AS idStandard,
s.name AS standardName

FROM standard s;

END
```

Depends On 2



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	12/6/2024 22:38:42
Last Modified	12/6/2024 22:38:42

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectStandardsJSON ()

DETERMINISTIC

BEGIN

SELECT

JSON_ARRAYAGG(JSON_OBJECT('key', s.id,
    'standard', s.name,
    'materials', (selectMaterials_has_Standard(s.id)),
    'enviroments', (selectEnviroments_has_Standard(s.id)),
    'endCaps', (selectEndCaps_has_Standard(s.id)),
    'conditionalPeriods', (selectConditionalPeriods_has_Standard(s.id)),
    'testTypes', (selectTest_Type(s.id)))) AS `standards`

FROM standard s;
```

Depends On 🕡

databaseManager@%

selectMaterials_has_Standard

selectEnviroments_has_Standard

selectEndCaps_has_Standard

selectConditionalPeriods_has_Standard

selectTest_Type

standard

Used By

selectStandarsComplete

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

SQL Script

```
PROCEDURE selectStandarsComplete ()
   s.id,
   s.name,
        'endcap', e.name))
     FROM endcap e
     WHERE e.standard = s.id) AS `endCaps`,
       JSON_ARRAYAGG(JSON_OBJECT('id', e.id,
       'insertFluid', e.inside,
        'outsideFluid', e.outside))
     FROM enviroment e
     WHERE e.standard = s.id) AS `enviroment`,
       JSON_ARRAYAGG(JSON_OBJECT('id', cp.id,
        'minwall', cp.minwall,
        'maxwall', cp.maxwall,
        'time', cp.time))
     FROM conditional_period cp
     WHERE cp.standard = s.id) AS conditionalPeriod,
       JSON_ARRAYAGG(JSON_OBJECT('id', mhs.id,
       'idMaterial', mhs.material,
       'material', m.name,
       'description', m.description))
     FROM material_has_standard mhs
        INNER JOIN material m
         ON mhs.material = m.id
     WHERE mhs.standard = s.id) AS relatedMaterial
  FROM standard s;
```

Depends On

enviroment
conditional_period
material_has_standard
material
standard

Used By
No items found



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE selectTestType (IN idStandard int UNSIGNED)

DETERMINISTIC

BEGIN

SELECT

tt.id,

tt.testType

FROM test_type tt

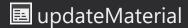
WHERE tt.standard = idStandard;

END
```

Depends On 2



Used By



Properties

Name	Value	
SQL Access	CONTAINS SQL	
Deterministic	True	
Security	Definer	
Definer	root	
SQL Mode	NO_AUTO_VALUE_ON_ZERO	
Language	SQL	
Created	11/6/2024 11:06:01	
Last Modified	11/6/2024 11:06:01	

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
namelN	VARCHAR	20		
descriptionIN	VARCHAR	80		

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE updateMaterial (IN id int UNSIGNED, IN nameIN varchar(20), IN descriptionIN varchar(80))

DETERMINISTIC

BEGIN

DECLARE result text;

IF (existMaterial(id) = TRUE) THEN

UPDATE LOW_PRIORITY material m

SET m.id = id,

m.name = nameIN,

m.description = descriptionIN

WHERE m.id = id;

SELECT

'Successful Updated!' INTO result;

ELSE

SELECT

'Unsuccessful Updated!' INTO result;

END IF;

SELECT

result AS `response`;
```

Depends On 3





Used By



Properties

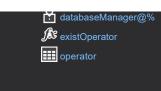
Name	Value	
SQL Access	CONTAINS SQL	
Deterministic	True	
Security	Definer	
Definer	root	
SQL Mode	NO_AUTO_VALUE_ON_ZERO	
Language	SQL	
Created	11/6/2024 11:06:03	
Last Modified	11/6/2024 11:06:03	

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
dni	INT		11	
name	VARCHAR	50		
familyname	VARCHAR	50		

SQL Script

Depends On 3



Used By

updateSettingsSpecification

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
Language	sqL
Created	27/6/2024 11:04:58
Last Modified	27/6/2024 11:04:58

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
timeIN	SMALLINT		6	
timeTypeIN	VARCHAR	1		
temperatureIN	TINYINT		4	

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE updateSettingsSpecification (IN id int UNSIGNED, IN timeIN smallint, IN timeTypeIN varchar(1), IN temperatureIN tinyint)

DETERMINISTIC

BEGIN

DECLARE result text;

IF (existSettingsSpecification(id) = TRUE) THEN

UPDATE LOW_PRIORITY settings_specification sc

SET sc.id = id,

sc.time = timeIN,
sc.timeType = timeTypeIN,
sc.temperature = temperatureIN

WHERE sc.id = id;

SELECT

'Successful Updated!' INTO result;

ELSE

SELECT

'Unsuccessful Updated!' INTO result;

END

SELECT

result AS response;

END
```

Depends On





Used By No items found



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:01
Last Modified	11/6/2024 11:06:01

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
namelN	VARCHAR	20		
descriptionIN	VARCHAR	80		

SQL Script

```
CREATE

DEFINER = 'root'

PROCEDURE updateSpecification (IN id int UNSIGNED, IN nameIN varchar(20), IN descriptionIN varchar(80))

DETERMINISTIC

BEGIN

DECLARE result text;

If (existSpecification(id) = TRUE) THEN

UPDATE LOW_PRIORITY specification s

SET s.id = id,

s.name = nameIN,
s.description = descriptionIN

WHERE s.id = id;

SELECT
    'Successful Updated!' INTO result;

ELSE

SELECT
    'Unsuccessful Updated!' INTO result;

END IF;

SELECT
    result AS `response`;

END
```

Depends On 3





Used By



Properties

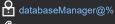
Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
namelN	VARCHAR	60		

SQL Script

Depends On 3



sexistStandard

standard

Used By No items found



Objects 19

Name	Description
existConditionalPeriod	
existConfiguration	
existEndCap	
existEnviroment	
<u>existMaterial</u>	
existOperator	
<u>existRelatedMaterial</u>	
<u>existSettingsSpecification</u>	
existSpecification	
<u>existSpecimen</u>	
<u>existStandard</u>	
existTest_Type	
selectConditionalPeriods_has_Standard	Selection of Conditional Periods of a Standard
selectEndCaps_has_Standard	
selectEnviroments_has_Standard	
selectMaterials_has_Standard	
selectSettingsSpecification	
<u>selectSpecifications</u>	
selectTest_Type	



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION existConditionalPeriod (id int UNSIGNED)
RETURNS tinyint(1)
DETERMINISTIC
BEGIN

DECLARE selected bigint(255);
SELECT
COUNT(*) INTO selected
FROM conditional_period cp
WHERE cp.id = id;

If selected != 1 THEN
RETURN FALSE;
ELSE
RETURN TRUE;
END IF;
```

Depends On 🕕

conditional_period

Used By 1

deleteConditionalPeriod



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:01
Last Modified	11/6/2024 11:06:01

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

Depends On 1

specification

Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION existEndCap (id int UNSIGNED)
RETURNS tinyint(1)
DETERMINISTIC
BEGIN

DECLARE selected bigint(255);
SELECT
COUNT(*) INTO selected
FROM endcap e
WHERE e.id = id;

If selected != 1 THEN
RETURN FALSE;
ELSE
RETURN TRUE;
END IF;
```

Depends On 🕕



Used By 1

deleteEndCap



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION existEnviroment (id int UNSIGNED)
RETURNS tinyint(1)
DETERMINISTIC
BEGIN

DECLARE selected bigint(255);
SELECT
COUNT(*) INTO selected
FROM enviroment e
WHERE e.id = id;

If selected != 1 THEN
RETURN FALSE;
ELSE
RETURN TRUE;
END IF;
```

Depends On 1







Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:01
Last Modified	11/6/2024 11:06:01

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION existMaterial (id int UNSIGNED)
RETURNS tinyint(1)
DETERMINISTIC
BEGIN

DECLARE selected bigint(255);
SELECT
COUNT(*) INTO selected
FROM material m
WHERE m.id = id;

If selected!= 1 THEN
RETURN FALSE;
ELSE
RETURN TRUE;
END IF;
```

Depends On •







Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:03
Last Modified	11/6/2024 11:06:03

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION existOperator (id int UNSIGNED)
RETURNS tinyint(1)
DETERMINISTIC
BEGIN

DECLARE selected bigint(255);
SELECT
COUNT(*) INTO selected
FROM operator o
WHERE o.id = id;

If selected!= 1 THEN
RETURN FALSE;
ELSE
RETURN TRUE;
END IF;
```

Depends On 1







Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION existRelatedMaterial (id int UNSIGNED)
RETURNS tinyint(1)
DETERMINISTIC
BEGIN

DECLARE selected bigint(255);
SELECT
COUNT(*) INTO selected
FROM material_has_standard mhs
WHERE mhs.id = id;

If selected != 1 THEN
RETURN FALSE;
ELSE
RETURN TRUE;
END IF;
```

Depends On •

material_has_standard

Used By 1

deleteRelatedMaterial

♠ existSettingsSpecification

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
Language	sqL
Created	27/6/2024 11:05:34
Last Modified	27/6/2024 11:05:34

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

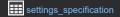
```
CREATE
DEFINER = 'root'
FUNCTION existSettingsSpecification (id int UNSIGNED)
RETURNS tinyint(1)
DETERMINISTIC
BEGIN

DECLARE selected bigint(255);

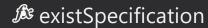
SELECT
    COUNT(*) INTO selected
    FROM settings_specification sc
    WHERE sc.id = id;

If selected != 1 THEN
    RETURN FALSE;
ELSE
    RETURN TRUE;
END IF;
```

Depends On 1







Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:01
Last Modified	11/6/2024 11:06:01

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION existSpecification (id int UNSIGNED)
RETURNS tinyint(1)
DETERMINISTIC
BEGIN

DECLARE selected int;
SELECT
COUNT(*) INTO selected
FROM specification s
WHERE s.id = id;

If selected != 1 THEN
RETURN FALSE;
ELSE
RETURN TRUE;
END IF;
```

Depends On 1







Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:01
Last Modified	11/6/2024 11:06:01

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

Depends On 1



Used By



Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

```
CREATE

DEFINER = 'root'

FUNCTION existStandard (id int UNSIGNED)

RETURNS tinyint(1)

DETERMINISTIC

BEGIN

DECLARE selected bigint(255);

SELECT

COUNT(*) INTO selected

FROM standard s

WHERE s.id = id;

If selected! = 1 THEN

RETURN FALSE;

ELSE

RETURN TRUE;
END IF;
```

Depends On 1







Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	sqL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
id	INT		11	
(Result)	TINYINT		1	

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION existTest_Type (id int UNSIGNED)
RETURNS tinyint(1)
DETERMINISTIC
BEGIN

DECLARE selected bigint(255);
SELECT
COUNT(*) INTO selected
FROM test_type tt
WHERE tt.id = id;

If selected != 1 THEN
RETURN FALSE;
ELSE
RETURN TRUE;
END IF;
```

Depends On 🕕

test_type

Used By 1

deleteTestType



Selection of Conditional Periods of a Standard

Properties

Name	Value	
SQL Access	CONTAINS SQL	
Deterministic	True	
Security	Definer	
Definer	root	
SQL Mode	NO_AUTO_VALUE_ON_ZERO	
Language	SQL	
Created	24/6/2024 23:05:15	
Last Modified	24/6/2024 23:05:15	

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
(Result)	JSON	0		

SQL Script

```
FUNCTION selectConditionalPeriods_has_Standard (idStandard int UNSIGNED)
 DECLARE result json;
 DECLARE elements tinyint(1);
   COUNT(*) INTO elements
  FROM conditional_period cp
 WHERE cp.standard = idStandard;
  IF (elements > 0) THEN
   WITH periodsOrder
       CONCAT(CONVERT(cp.time, char), ' ', cp.timeType, ' ± ', CONVERT(cp.aproxTime, char), ' ', cp.aproxType) AS
       `condPeriod`,
       cp.minwall,
       cp.maxwall
     FROM conditional_period cp
     WHERE cp.standard = idStandard)
     JSON_ARRAYAGG(JSON_OBJECT('key', po.`key`,
      'condPeriod', po.condPeriod,
      'minwall', po.minwall,
      'maxwall', po.maxwall)) INTO result
```

```
FROM periodsOrder po;

ELSE
SELECT
'[]' INTO result;
END IF;

RETURN result;
END

Depends On 
Conditional_period

Used By 
SelectStandardsJSON
```

selectEndCaps_has_Standard

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Language	SQL
Created	11/6/2024 11:06:02
Last Modified	11/6/2024 11:06:02

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
(Result)	JSON	0		

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION selectEndCaps_has_Standard (idStandard int UNSIGNED)
RETURNS json
DETERMINISTIC
BEGIN

DECLARE result json;
DECLARE elements tinyint(1);

SELECT
COUNT(*) INTO elements
FROM endcap cc
WHERE ec.standard = idStandard;

If (elements > 0) THEN
SELECT
JSON_ARRAYAGG(JSON_OBJECT('key', ec.id,
    'endcap', ec.name)) INTO result
FROM endcap ec
WHERE ec.standard = idStandard;

ELSE
SELECT
'[]' INTO result;
END IF;

RETURN result;
```

Depends On 🛈





selectStandardsJSON

selectEnviroments_has_Standard

Description

Properties

Name	Value	
SQL Access	CONTAINS SQL	
Deterministic	True	
Security	Definer	
Definer	root	
SQL Mode	NO_AUTO_VALUE_ON_ZERO	
Language	sqL	
Created	11/6/2024 11:06:02	
Last Modified	11/6/2024 11:06:02	

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
(Result)	JSON	0		

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION selectEnviroments_has_Standard (idStandard int UNSIGNED)
RETURNS json
DETERMINISTIC
BEGIN

DECLARE result json;
DECLARE elements tinyint;

SELECT
COUNT(*) INTO elements
FROM enviroment e
WHERE e.standard = idStandard;

IF (elements > 0) THEN
SELECT
JSON_ARRAYAGG(JSON_OBJECT('key', e.id,
    'insidefluid', e.inside,
    'outsideFluid', e.outside)) INTO result
FROM enviroment e
WHERE e.standard = idStandard;
ELSE
SELECT
'[]' INTO result;
END IF;

RETURN result;
```

Depends On 🕕



Used By 🛈

selectStandardsJSON



Description

Properties

Name	Value	
SQL Access	CONTAINS SQL	
Deterministic	True	
Security	Definer	
Definer	root	
SQL Mode	NO_AUTO_VALUE_ON_ZERO	
Language	sqL	
Created	24/6/2024 17:13:33	
Last Modified	24/6/2024 17:13:33	

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
(Result)	JSON	0		

```
FUNCTION selectMaterials_has_Standard (idStandard int UNSIGNED)
 DECLARE result json;
 DECLARE elements tinyint;
   COUNT(*) INTO elements
 FROM material_has_standard mhs
 WHERE mhs.standard = idStandard;
 IF (elements > 0) THEN
     JSON_ARRAYAGG(JSON_OBJECT('key', mhs.id,
      'idMaterial', mhs.material,
      'material', m.name,
     'descriptioin', m.description)) INTO result
   FROM material_has_standard mhs
     INNER JOIN material m
       ON m.id = mhs.material
   WHERE mhs.standard = idStandard;
     '[]' INTO result;
 RETURN result;
```

Depends On ②

iiii material_has_standard
iiii material

Used By ③

selectStandardsJSON

ℬ selectSettingsSpecification

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
Language	sqL
Created	27/6/2024 11:05:10
Last Modified	27/6/2024 11:05:10

Parameters

Name	Data Type	Length	Precision	Scale
idSpecification	INT		11	
(Result)	JSON	0		

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION selectSettingsSpecification (idSpecification int UNSIGNED)
RETURNS json
DETERMINISTIC
BEGIN

DECLARE result json;
DECLARE elements tinyint;

SELECT
COUNT(*) INTO elements
FROM settings_specification sc
WHERE sc.specification = idSpecification;

IF (elements > 0) THEN
SELECT
JSON_ARRAYAGG(JSON_OBJECT('key', sc.id,
    'time', sc.time,
    'type', sc.timeType,
    'tymey', sc.timeType,
    'temperature', sc.temperature)) INTO result
FROM settings_specification sc
WHERE sc.specification = idSpecification;
ELSE
SELECT
'[]' INTO result;
END IF;

RETURN result;
```

Depends On 1



Used By

selectSpecifications

Description

Properties

Name	Value
SQL Access	CONTAINS SQL
Deterministic	True
Security	Definer
Definer	root
SQL Mode	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
Language	sqL
Created	27/6/2024 11:05:10
Last Modified	27/6/2024 11:05:10

Parameters

Name	Data Type	Length	Precision	Scale
idMaterial	INT		11	
(Result)	JSON	0		

SQL Script

```
CREATE
DEFINER = 'root'
FUNCTION selectSpecifications (idMaterial int UNSIGNED)
RETURNS json
DETERMINISTIC
BEGIN

DECLARE result json;
DECLARE elements tinyint;

SELECT
COUNT(*) INTO elements
FROM specification s
WHERE s.material = idMaterial;

IF (elements > 0) THEN
SELECT
JSON_ARRAYAGG(JSON_OBJECT('key', s.id,
    'specification', s.name,
    'description', (IF(s.description IS NULL, '', s.description)),
    'configurations', (selectSettingsSpecification(s.id)))) INTO result
FROM specification s
WHERE s.material = idMaterial;
ELSE
SELECT
    '[]' INTO result;
END IF;
RETURN result;
```

Depends On 2



Used By No items found



Description

Properties

Name	Value	
SQL Access	CONTAINS SQL	
Deterministic	True	
Security	Definer	
Definer	root	
SQL Mode	NO_AUTO_VALUE_ON_ZERO	
Language	sqL	
Created	25/6/2024 01:20:43	
Last Modified	25/6/2024 01:20:43	

Parameters

Name	Data Type	Length	Precision	Scale
idStandard	INT		11	
(Result)	JSON	0		

SQL Script

```
CREATE

DEFINER = 'root'
FUNCTION selectTest_Type (idStandard int UNSIGNED)
RETURNS json
DETERMINISTIC
BEGIN

DECLARE result json;
DECLARE elements tinyint;

SELECT
    COUNT(*) INTO elements
FROM test_type tt
WHERE tt.standard = idStandard;

If (elements > 0) THEN
SELECT
    JSON_ARRAYAGG(JSON_OBJECT('key', tt.id,
    'testtype', tt.testType)) INTO result
FROM test_type tt
WHERE tt.standard = idStandard;

ELSE
SELECT
    '[]' INTO result;
END IF;

RETURN result;
```

Depends On 1





selectStandardsJSON

Project / Servers / Local_Development / Databases / static_db / Triggers

Clippers

Objects
Name

deleteConfigurationsChilds

deleteStandardChilds

deleteStandardChilds

deleteConfigurationsChilds

Properties

Name	Value
Definer	root
SQL Mode	ONLY_FULL_GROUP_BY,STRICT_TRANS_TABLES,NO_ZERO_IN_DATE,NO_ZERO_DATE,ERROR_FOR_DIVISION_BY_ZERO,NO_ENGINE_SUBSTITUTION
Event	DELETE
Timing	BEFORE
Follows	
Precedes	
Created	27/6/2024 11:01:28

SQL Script

```
CREATE
DEFINER = 'root'
TRIGGER deleteConfigurationsChilds
BEFORE DELETE
ON specification
FOR EACH ROW
BEGIN

DECLARE childsCount int UNSIGNED;

SELECT
COUNT(*) INTO childsCount
FROM settings_specification sc
WHERE sc.specification = OLD.id;

IF childsCount > 0 THEN
DELETE LOW_PRIORITY QUICK
FROM settings_specification sc
WHERE sc.specification = OLD.id;

END IF;
```

Depends On 2

specification

settings_specification

Used By

No items found

deleteMaterialChilds

Properties

Name	Value
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Event	DELETE
Timing	BEFORE
Follows	
Precedes	
Created	11/6/2024 11:06:04

SQL Script

```
CREATE
DEFINER = 'root'
TRIGGER deleteMaterialChilds
BEFORE DELETE
ON material
FOR EACH ROW
BEGIN

DECLARE childsCount int UNSIGNED;

SELECT
COUNT(*) INTO childsCount
FROM specification s
WHERE s.material = OLD.id;

If childsCount > 0 THEN
DELETE LOW PRIORITY QUICK
FROM specification s
WHERE s.material = OLD.id;

END IF;

SELECT
COUNT(*) INTO childsCount
FROM material | has_standard mhs
WHERE mhs.material = OLD.id;

If childsCount > 0 THEN
DELETE LOW_PRIORITY QUICK
FROM material = OLD.id;

END IF;

SELECT
COUNT(*) INTO childsCount
FROM material = OLD.id;

If childsCount > 0 THEN
DELETE LOW_PRIORITY QUICK
FROM material | bas_standard mhs
WHERE mhs.material = OLD.id;

END IF;

END
```

Depends On

material
specification
material_has_standard

Used By

No items found

deleteStandardChilds

Properties

Name	Value
Definer	root
SQL Mode	NO_AUTO_VALUE_ON_ZERO
Event	DELETE
Timing	BEFORE
Follows	
Precedes	
Created	11/6/2024 11:06:04

```
TRIGGER deleteStandardChilds
BEFORE DELETE
ON standard
 DECLARE elements int UNSIGNED;
   COUNT(*) INTO elements
 FROM endcap e
 WHERE e.standard = OLD.id;
 IF elements > 0 THEN
     FROM endcap e
   WHERE e.standard = OLD.id;
   COUNT(*) INTO elements
 FROM conditional_period cp
 WHERE cp.standard = OLD.id;
 IF elements > 0 THEN
     FROM conditional_period cp
   WHERE cp.standard = OLD.id;
   COUNT(*) INTO elements
 FROM enviroment e
 WHERE e.standard = OLD.id;
 IF elements > 0 THEN
     FROM enviroment e
   WHERE e.standard = OLD.id;
   COUNT(*) INTO elements
 FROM test_type tt
  WHERE tt.standard = OLD.id;
  IF elements > 0 THEN
```

```
DELETE LOW_PRIORITY QUICK
   FROM test_type tt

WHERE tt.standard = OLD.id;
END IF;

SELECT
   COUNT(*) INTO elements
FROM material_has_standard mhs
WHERE mhs.standard = OLD.id;

IF elements > 0 THEN
   DELETE LOW_PRIORITY QUICK
   FROM material_has_standard mhs
WHERE mhs.standard = OLD.id;

END IF;

END
```

Depends On

standard

endcap

conditional_period

enviroment

test_type

material_has_standard

Used By

No items found



Objects 3

dataCollector@%
databaseManager@%

reportGenerator@%

dataCollector@%

Object Level Permissions

Туре	Action	Object Name	Object Type
Grant	EXECUTE	countspecimens	Procedure
Grant	EXECUTE	insertdata	Procedure
Grant	EXECUTE	insertsample	Procedure
Grant	EXECUTE	insertspecimen	Procedure
Grant	EXECUTE	selectconditionalperiods	Procedure
Grant	EXECUTE	selectendcaps	Procedure
Grant	EXECUTE	selectenviroment	Procedure
Grant	EXECUTE	selectmaterials	Procedure
Grant	EXECUTE	selectoperators	Procedure
Grant	EXECUTE	selectsample	Procedure
Grant	EXECUTE	selectsampleid	Procedure
Grant	EXECUTE	selectsettingsspecification	Procedure
Grant	EXECUTE	selectspecifications	Procedure
Grant	EXECUTE	selectspecimen	Procedure
Grant	EXECUTE	selectstandards	Procedure
Grant	EXECUTE	selecttesttype	Procedure

```
CREATE USER 'dataCollector'@'%' IDENTIFIED WITH caching_sha2_password PASSWORD EXPIRE NEVER;
GRANT EXECUTE ON PROCEDURE data_db.countspecimens TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE data_db.insertdata TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE data_db.insertsample TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE data_db.insertspecimen TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectconditionalperiods TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectendcaps TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectenviroment TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectmaterials TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectoperators TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE data_db.selectsample TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE data_db.selectsampleid TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectsettingsspecification TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectspecifications TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE data_db.selectspecimen TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectstandards TO 'dataCollector'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selecttesttype TO 'dataCollector'@'%';
```

databaseManager@%

Object Level Permissions 29

Туре	Action	Object Name	Object Type
Grant	EXECUTE	deleteconditionalperiod	Procedure
Grant	EXECUTE	deleteendcap	Procedure
Grant	EXECUTE	deleteenviroment	Procedure
Grant	EXECUTE	deletematerial	Procedure
Grant	EXECUTE	deleteoperator	Procedure
Grant	EXECUTE	deleterelatedmaterial	Procedure
Grant	EXECUTE	deletesettingsspecification	Procedure
Grant	EXECUTE	deletespecification	Procedure
Grant	EXECUTE	deletestandard	Procedure
Grant	EXECUTE	deletetesttype	Procedure
Grant	EXECUTE	insertconditionalperiod	Procedure
Grant	EXECUTE	insertendcap	Procedure
Grant	EXECUTE	insertenviroment	Procedure
Grant	EXECUTE	insertmaterial	Procedure
Grant	EXECUTE	insertoperator	Procedure
Grant	EXECUTE	insertrelatedmaterial	Procedure
Grant	EXECUTE	insertsettingsspecification	Procedure
Grant	EXECUTE	insertspecification	Procedure
Grant	EXECUTE	insertstandard	Procedure
Grant	EXECUTE	inserttesttype	Procedure
Grant	EXECUTE	selectmaterialsjson	Procedure
Grant	EXECUTE	selectoperatorsjson	Procedure
Grant	EXECUTE	selectstandardsjson	Procedure
Grant	EXECUTE	updatematerial	Procedure
Grant	EXECUTE	updateoperator	Procedure
Grant	EXECUTE	updatesettingsspecification	Procedure
Grant	EXECUTE	updatespecification	Procedure
Grant	EXECUTE	updatestandard	Procedure
Grant	SELECT	material	Table

```
CREATE USER 'databaseManager'@'%' IDENTIFIED WITH caching_sha2_password PASSWORD EXPIRE NEVER;
GRANT EXECUTE ON PROCEDURE static_db.deleteconditionalperiod TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.deleteendcap TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.deleteenviroment TO 'databaseManager'@'%';
```

```
GRANT EXECUTE ON PROCEDURE static_db.deletematerial TO 'databaseManager'@
\label{lem:grant_execute} \textit{GRANT} \ \textit{EXECUTE} \ \textit{ON} \ \textit{PROCEDURE} \ \textit{static\_db.deleteoperator} \ \textit{TO} \ '\textit{databaseManager'} @'\%';
GRANT EXECUTE ON PROCEDURE static_db.deleterelatedmaterial TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.deletesettingsspecification TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.deletespecification TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.deletestandard TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.deletetesttype TO 'databaseManager'
GRANT EXECUTE ON PROCEDURE static_db.insertconditionalperiod TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.insertendcap TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.insertenviroment TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.insertmaterial TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.insertoperator TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.insertrelatedmaterial TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.insertsettingsspecification TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.insertspecification TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.insertstandard TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.inserttesttype TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectmaterialsjson TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectoperatorsjson TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.selectstandardsjson TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.updatematerial TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.updateoperator TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.updatesettingsspecification TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.updatespecification TO 'databaseManager'@'%';
GRANT EXECUTE ON PROCEDURE static_db.updatestandard TO 'databaseManager'@'%';
GRANT SELECT ON TABLE static_db.material TO 'databaseManager'@'%';
```

reportGenerator@%

Object Level Permissions

Туре	Action	Object Name	Object Type
Grant	EXECUTE	static_db	Database
Grant	EXECUTE	deletetest	Procedure
Grant	EXECUTE	selectcomparetests	Procedure
Grant	EXECUTE	selecttestdata	Procedure
Grant	EXECUTE	selecttest	Procedure
Grant	EXECUTE	selecttests	Procedure
Grant	EXECUTE	updatespecimen	Procedure

```
CREATE USER 'reportGenerator'@'%' IDENTIFIED WITH caching_sha2_password PASSWORD EXPIRE NEVER;

GRANT EXECUTE ON *.* TO 'reportGenerator'@'%';

GRANT EXECUTE ON static_db.* TO 'reportGenerator'@'%';

GRANT EXECUTE ON PROCEDURE data_db.deletetest TO 'reportGenerator'@'%';

GRANT EXECUTE ON PROCEDURE data_db.selectcomparetests TO 'reportGenerator'@'%';

GRANT EXECUTE ON PROCEDURE data_db.selecttest TO 'reportGenerator'@'%';

GRANT EXECUTE ON PROCEDURE data_db.selecttestdata TO 'reportGenerator'@'%';

GRANT EXECUTE ON PROCEDURE data_db.selecttests TO 'reportGenerator'@'%';

GRANT EXECUTE ON PROCEDURE data_db.selecttests TO 'reportGenerator'@'%';

GRANT EXECUTE ON PROCEDURE data_db.updatespecimen TO 'reportGenerator'@'%';
```