Aufgabenbezogenen Auswertungen

Alle offenen Aufgaben eines bestimmten Users

```
SELECT i.lastname AS Nachname, i.firstname AS Vorname, i."name" AS Username,
t."name" AS Task, vi."name" AS "Instanzname", vi.definitionName AS
Prozessname
FROM view_activity a, view_task t, view_identity i, view_instance vi
WHERE a.id = t.activity AND t.actor = i.id AND a."end" IS NULL AND t."end"
IS NULL AND vi.id = a.instanceId AND vi.archiv = 0 AND vi.END IS NULL AND
i.lastname = 'NACHNAME'
```

Anzahl an erstellten, erledigten und offenen Aufgaben je Prozessdefinition

```
SELECT inst.definitionname AS Prozess,
COUNT(t1.id) AS "Anzahl erstellte Aufgaben",
COUNT(t2.id) AS "Anzahl erledigte Aufgaben",
COUNT(t3.id) AS "Anzahl offene Aufgaben"
FROM view_task t1 INNER JOIN view_activity act ON t1.activity=act.id
INNER JOIN view_instance inst ON t1.instanceid = inst.id
LEFT JOIN view_task t2 ON t1.id=t2.id AND (t2."end" IS NOT NULL OR act."end"
IS NOT NULL)
LEFT JOIN view_task t3 ON t1.id=t3.id AND t3."end" IS NULL AND inst.END IS
NULL AND inst.archiv = 0
GROUP BY inst.definitionname
```

Anzahl an erstellten, erledigten und offenen Aufgaben einer bestimmten Prozessdefinition

```
SELECT inst.definitionname AS Prozess,
COUNT(t1.id) AS "Anzahl erstellte Aufgaben",
COUNT(t2.id) AS "Anzahl erledigte Aufgaben",
COUNT(t3.id) AS "Anzahl offene Aufgaben"
FROM view_task t1 INNER JOIN view_activity act ON t1.activity=act.id INNER
JOIN view_instance inst ON t1.instanceid = inst.id
LEFT JOIN view_task t2 ON t1.id=t2.id AND (t2."end" IS NOT NULL OR act."end"
IS NOT NULL)
LEFT JOIN view_task t3 ON t1.id=t3.id AND t3."end" IS NULL AND inst.END IS
NULL AND inst.archiv = 0
WHERE inst.definitionname ='PROZESSDEFINITIONSNAME'
  GROUP BY inst.definitionname
```

Durchschnittliche Zeit (in hh:mm:ss) von Erstellung bis Erledigung von allen

Aufgaben (egal von welcher Prozessdefinition und auch außerhalb der Arbeitszeit)

```
SELECT
CASE
WHEN FLOOR(sekunden)<60
THEN
  CONCAT(CASE WHEN FLOOR(sekunden)<10 THEN  '00:00:0' ELSE '00:00:'
END,FLOOR(sekunden))
WHEN FLOOR(sekunden)>3600
THEN
 CONCAT(CONCAT(CONCAT(CONCAT(h, CASE WHEN FLOOR((FLOOR(sekunden)- h*3600-
sek)/60)<10 THEN ':0' ELSE ':' END),FLOOR((FLOOR(sekunden)- h*3600-
sek)/60)),CASE WHEN sek < 10 THEN ':0' ELSE ':' END),sek)
ELSE
CONCAT(CONCAT(CONCAT(CASE WHEN FLOOR(sekunden/60)<10 THEN '00:0' ELSE '00:'
END,FLOOR(sekunden/60)), CASE WHEN FLOOR(sek)<10 THEN ':0' ELSE ':' END),
FLOOR(sek))
END
AS "Durchschn. LZ v. Aufg."
FROM (
SELECT
FLOOR(AVG((TO_NUMBER (TO_CHAR ("end", 'J')) - TO_NUMBER (TO_CHAR ("start",
'J'))) * 86400 + ( TO_NUMBER (TO_CHAR ("end", 'SSSSS')) - TO_NUMBER (TO_CHAR
("start", 'SSSSS'))))) AS sekunden,
FLOOR(MOD(AVG((TO_NUMBER (TO_CHAR ("end", 'J')) - TO_NUMBER (TO_CHAR
("start", 'J'))) * 86400 + ( TO_NUMBER (TO_CHAR ("end", 'SSSSS')) -
TO_NUMBER (TO_CHAR ("start", 'SSSSS')))),60)) AS sek,
FLOOR(AVG((TO_NUMBER (TO_CHAR ("end", 'J')) - TO_NUMBER (TO_CHAR ("start",
'J'))) * 86400 + ( TO_NUMBER (TO_CHAR ("end", 'SSSSS')) - TO_NUMBER (TO_CHAR
("start", 'SSSSS'))))/3600) AS h
FROM view_task WHERE "end" IS NOT NULL
)
```

Durchschnittliche Zeit (in hh:mm:ss) von Erstellung bis Erledigung von allen Aufgaben je Monat (egal von welcher Prozessdefinition und auch außerhalb der Arbeitszeit)

```
SELECT Monat,
CASE
WHEN FLOOR(sekunden)<60
THEN
CONCAT(CASE WHEN FLOOR(sekunden)<10 THEN  '00:00:0' ELSE '00:00:'
END,FLOOR(sekunden))
WHEN FLOOR(sekunden)>3600
THEN
CONCAT(CONCAT(CONCAT(CONCAT(h, CASE WHEN FLOOR((FLOOR(sekunden)- h*3600-
```

```
sek)/60)<10 THEN ':0' ELSE ':' END),FLOOR((FLOOR(sekunden)- h*3600-
sek)/60)),CASE WHEN sek < 10 THEN ':0' ELSE ':' END),sek)
ELSE
CONCAT(CONCAT(CONCAT(CASE WHEN FLOOR(sekunden/60)<10 THEN '00:0' ELSE '00:'
END,FLOOR(sekunden/60)), CASE WHEN FLOOR(sek)<10 THEN ':0' ELSE ':' END),
FLOOR(sek))
END
AS "Durchschn. LZ v. Aufg."
FROM
(
SELECT
CONCAT(CONCAT(SUBSTR(to_char(to_date(SUBSTR("start",0,(INSTR("start", '.',
-1)-1)), 'DD-MM-YY HH24:MI:SS'), 'Month'),1,3),' '),EXTRACT(YEAR FROM
to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS'))) AS Monat,
FLOOR(AVG((TO_NUMBER (TO_CHAR ("end", 'J')) - TO_NUMBER (TO_CHAR ("start",
'J'))) * 86400 + ( TO_NUMBER (TO_CHAR ("end", 'SSSSS')) - TO_NUMBER (TO_CHAR
("start", 'SSSSS'))))) AS sekunden,
FLOOR(MOD(AVG((TO_NUMBER (TO_CHAR ("end", 'J')) - TO_NUMBER (TO_CHAR
("start", 'J'))) * 86400 + ( TO_NUMBER (TO_CHAR ("end", 'SSSSS')) -
TO_NUMBER (TO_CHAR ("start", 'SSSSS')))),60)) AS sek,
FLOOR(AVG((TO_NUMBER (TO_CHAR ("end", 'J')) - TO_NUMBER (TO_CHAR ("start",
'J'))) * 86400 + ( TO_NUMBER (TO_CHAR ("end", 'SSSSS')) - TO_NUMBER (TO_CHAR
("start", 'SSSSS'))))/3600) AS h,
EXTRACT(YEAR FROM to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')) AS j,
EXTRACT(MONTH FROM to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')) AS m
FROM view_task
WHERE isOpen = '0'
GROUP BY
EXTRACT(YEAR FROM to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')),
CONCAT(CONCAT(SUBSTR(to_char(to_date(SUBSTR("start",0,(INSTR("start", '.',
-1)-1)), 'DD-MM-YY HH24:MI:SS'), 'Month'),1,3),' '),EXTRACT(YEAR FROM
to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS'))),
EXTRACT(YEAR FROM to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')),
EXTRACT(MONTH FROM to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS'))
)
ORDER BY
j, m
```

Durchschnittliche Zeit (in hh:mm:ss) von Erstellung bis Erledigung von allen Aufgaben je Monat ohne Ausreißer (egal von welcher Prozessdefinition und auch außerhalb der Arbeitszeit)

```
SELECT
Monat,
CASE
WHEN FLOOR(sekunden)<60
THEN
  CONCAT(CASE WHEN FLOOR(sekunden)<10 THEN  '00:00:0' ELSE '00:00:'
END,FLOOR(sekunden))
WHEN FLOOR(sekunden)>3600
THEN
 CONCAT(CONCAT(CONCAT(CONCAT(h, CASE WHEN FLOOR((FLOOR(sekunden)- h*3600-
sek)/60)<10 THEN ':0' ELSE ':' END),FLOOR((FLOOR(sekunden)- h*3600-
sek)/60)),CASE WHEN sek < 10 THEN ':0' ELSE ':' END),sek)
ELSE
CONCAT(CONCAT(CONCAT(CASE WHEN FLOOR(sekunden/60)<10 THEN '00:0' ELSE '00:'
END,FLOOR(sekunden/60)), CASE WHEN FLOOR(sek)<10 THEN ':0' ELSE ':' END),
FLOOR(sek))
END
AS "Durchschn. LZ v. Aufg."
FROM
(
SELECT
CONCAT(CONCAT(SUBSTR(to_char(to_date(SUBSTR("start",0,(INSTR("start", '.',
-1)-1)), 'DD-MM-YY HH24:MI:SS'), 'Month'),1,3),' '),EXTRACT(YEAR FROM
to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS'))) AS Monat,
FLOOR(AVG((TO_NUMBER (TO_CHAR ("end", 'J')) - TO_NUMBER (TO_CHAR ("start",
'J'))) * 86400 + ( TO_NUMBER (TO_CHAR ("end", 'SSSSS')) - TO_NUMBER (TO_CHAR
("start", 'SSSSS'))))) AS sekunden,
FLOOR(MOD(AVG((TO_NUMBER (TO_CHAR ("end", 'J')) - TO_NUMBER (TO_CHAR
("start", 'J'))) * 86400 + ( TO_NUMBER (TO_CHAR ("end", 'SSSSS')) -
TO_NUMBER (TO_CHAR ("start", 'SSSSS')))),60)) AS sek,
FLOOR(AVG((TO_NUMBER (TO_CHAR ("end", 'J')) - TO_NUMBER (TO_CHAR ("start",
'J'))) * 86400 + ( TO_NUMBER (TO_CHAR ("end", 'SSSSS')) - TO_NUMBER (TO_CHAR
("start", 'SSSSS'))))/3600) AS h
FROM view_task
WHERE isOpen = '0'
AND
(((to_date(SUBSTR("end",0,(INSTR("end", '.', -1)-1)), 'DD-MM-YY HH24:MI:SS')
- DATE '1970-01-01' ) * 60 * 60 * 24) -
((to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)) < 10000
GROUP BY
EXTRACT(YEAR FROM to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')),
CONCAT(CONCAT(SUBSTR(to_char(to_date(SUBSTR("start",0,(INSTR("start", '.',
-1)-1)), 'DD-MM-YY HH24:MI:SS'), 'Month'),1,3),' '),EXTRACT(YEAR FROM
to_date(SUBSTR("start",0,(INSTR("start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')))
```

```
)
```

Alle offenen Aufgaben mit aktiver Instanz (nicht beendet und nicht archiviert) und aktiver Aktivität (nicht beendet und nicht archiviert) mit Bearbeiter

```
SELECT tsk."name" AS Aufgabe,
inst."name" AS "Instanz",
CONCAT(CONCAT(CONCAT(CONCAT(CONCAT(ident.firstname, ' '), ident.lastname),'
('), grouped."name"),')') AS "Bearbeiter",
def."name" AS "Prozess",
tsk.creationTime AS "Erstellzeit"
FROM view_task tsk
LEFT JOIN view_identity ident ON tsk.actor = ident.id
LEFT JOIN view_identity grouped ON tsk.pooledActor = grouped.id
INNER JOIN view_instance inst ON tsk.instanceId = inst.id
INNER JOIN view_definition def ON inst.definitionId = def.id
INNER JOIN view_activity act ON tsk.activity = act.id
WHERE inst.archiv = '0' AND tsk.isOpen = '1'
```

Durchschnittliche, minimale und maximale Durchlaufszeit aller Aufgaben einer bestimmten Prozessdefinition (in hh:mm:ss)

```
SELECT
name1 AS "Aufgabe",
CASE
WHEN FLOOR(avgSekunden)<60
THEN
  CONCAT(CASE WHEN FLOOR(avgSekunden)<10 THEN '00:00:0' ELSE '00:00:'
END,FLOOR(avgSekunden))
WHEN FLOOR(avgSekunden)>3600
THEN
  CONCAT(CONCAT(CONCAT(CONCAT(avgH, CASE WHEN FLOOR((FLOOR(avgSekunden)-
avgH*3600-avgSek)/60)<10 THEN ':0' ELSE ':' END), FLOOR((FLOOR(avgSekunden)-
avgH*3600-avgSek)/60)), CASE WHEN avgSek<10 THEN ':0' ELSE ':' END), avgSek)
ELSE
  CONCAT(CONCAT(CONCAT(CASE WHEN FLOOR(avgSekunden/60)<10 THEN '00:0' ELSE
'00:' END,FLOOR(avgSekunden/60)), CASE WHEN FLOOR(avgSek)<10 THEN ':0' ELSE
':' END), FLOOR(avgSek))
END
AS "Durschn. DLZ",

CASE
WHEN FLOOR(minSekunden)<60
THEN
  CONCAT(CASE WHEN FLOOR(minSekunden)<10 THEN '00:00:0' ELSE '00:00:' END,
FLOOR(minSekunden))
WHEN FLOOR(minSekunden)>3600
```

```
THEN
  CONCAT(CONCAT(CONCAT(CONCAT(minH,CASE WHEN FLOOR((FLOOR(minSekunden)-
minH*3600-minSek)/60)<10 THEN ':0' ELSE ':' END),FLOOR((FLOOR(minSekunden)-
minH*3600-minSek)/60)),CASE WHEN minSek<10 THEN ':0' ELSE ':' END) ,minSek)
ELSE
  CONCAT(CONCAT(CONCAT(CASE WHEN FLOOR(minSekunden/60)<10 THEN '00:0' ELSE
'00:' END,FLOOR(minSekunden/60)),CASE WHEN FLOOR(minSek)<10 THEN ':0' ELSE
':' END),FLOOR(minSek))
END AS "Min. DLZ",

CASE
WHEN FLOOR(maxSekunden)<60
THEN
  CONCAT(CASE WHEN FLOOR(maxSekunden)<10 THEN '00:00:0' ELSE '00:00:' END,
FLOOR(maxSekunden))
WHEN FLOOR(maxSekunden)>3600
THEN
  CONCAT(CONCAT(CONCAT(CONCAT(maxH,CASE WHEN FLOOR((FLOOR(maxSekunden)-
maxH*3600-maxSek)/60)<10 THEN ':0' ELSE ':' END),FLOOR((FLOOR(maxSekunden)-
maxH*3600-maxSek)/60)),CASE WHEN maxSek<10 THEN ':0' ELSE ':' END) ,maxSek)
ELSE
  CONCAT(CONCAT(CONCAT(CASE WHEN FLOOR(maxSekunden/60)<10 THEN '00:0' ELSE
'00:' END,FLOOR(maxSekunden/60)),CASE WHEN FLOOR(maxSek)<10 THEN ':0' ELSE
':' END),FLOOR(maxSek))
END AS "Max. DLZ"

FROM
(

SELECT name1, minSekunden, minSek, minH, maxSekunden, maxSek, maxH,
avgSekunden, avgSek, avgH
FROM
(
SELECT FLOOR(MIN((
((to_date(SUBSTR(ta."end",0,(INSTR(ta."end", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
-
((to_date(SUBSTR(ta."start",0,(INSTR(ta."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
))) AS minSekunden,
ta."name" AS name1
FROM view_task ta, view_instance vi
WHERE vi.definitionName = 'PROZESSDEFINITIONSNAME'
AND vi.id = ta.instanceId
AND ta."end" IS NOT NULL
GROUP BY ta."name"
),
(
SELECT FLOOR(MOD(MIN((
```

```
((to_date(SUBSTR(ta."end",0,(INSTR(ta."end", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
-
((to_date(SUBSTR(ta."start",0,(INSTR(ta."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
)),60)) AS minSek,
ta."name" AS name2
FROM view_task ta, view_instance vi
WHERE vi.definitionName = 'PROZESSDEFINITIONSNAME'
AND vi.id = ta.instanceId
AND ta."end" IS NOT NULL
GROUP BY ta."name"
),
(
SELECT FLOOR(MIN((
((to_date(SUBSTR(ta."end",0,(INSTR(ta."end", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
-
((to_date(SUBSTR(ta."start",0,(INSTR(ta."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
)/3600)) AS minH,
ta."name" AS name3
FROM view_task ta, view_instance vi
WHERE vi.definitionName = 'PROZESSDEFINITIONSNAME'
AND vi.id = ta.instanceId
AND ta."end" IS NOT NULL
GROUP BY ta."name"
),
(
SELECT FLOOR(MAX((
((to_date(SUBSTR(ta."end",0,(INSTR(ta."end", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
-
((to_date(SUBSTR(ta."start",0,(INSTR(ta."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
))) AS maxSekunden,
ta."name" AS name4
FROM view_task ta, view_instance vi
WHERE vi.definitionName = 'PROZESSDEFINITIONSNAME'
AND vi.id = ta.instanceId
AND ta."end" IS NOT NULL
GROUP BY ta."name"
),
(
SELECT FLOOR(MOD(MAX((
((to_date(SUBSTR(ta."end",0,(INSTR(ta."end", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
-
((to_date(SUBSTR(ta."start",0,(INSTR(ta."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
)),60)) AS maxSek,
```

```
ta."name" AS name5
FROM view_task ta, view_instance vi
WHERE vi.definitionName = 'PROZESSDEFINITIONSNAME'
AND vi.id = ta.instanceId
AND ta."end" IS NOT NULL
GROUP BY ta."name"
),
(
SELECT FLOOR(MAX((
((to_date(SUBSTR(ta."end",0,(INSTR(ta."end", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
-
((to_date(SUBSTR(ta."start",0,(INSTR(ta."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
)/3600)) AS maxH,
ta."name" AS name6
FROM view_task ta, view_instance vi
WHERE vi.definitionName = 'PROZESSDEFINITIONSNAME'
AND vi.id = ta.instanceId
AND ta."end" IS NOT NULL
GROUP BY ta."name"
),
(
SELECT FLOOR(avg((
((to_date(SUBSTR(ta."end",0,(INSTR(ta."end", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
-
((to_date(SUBSTR(ta."start",0,(INSTR(ta."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
))) AS avgSekunden,
ta."name" AS name7
FROM view_task ta, view_instance vi
WHERE vi.definitionName = 'PROZESSDEFINITIONSNAME'
AND vi.id = ta.instanceId
AND ta."end" IS NOT NULL
GROUP BY ta."name"
),
(
SELECT FLOOR(MOD(avg((
((to_date(SUBSTR(ta."end",0,(INSTR(ta."end", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
-
((to_date(SUBSTR(ta."start",0,(INSTR(ta."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
)),60)) AS avgSek,
ta."name" AS name8
FROM view_task ta, view_instance vi
WHERE vi.definitionName = 'PROZESSDEFINITIONSNAME'
AND vi.id = ta.instanceId
```

```
AND ta."end" IS NOT NULL
GROUP BY ta."name"
),
(
SELECT FLOOR(avg((
((to_date(SUBSTR(ta."end",0,(INSTR(ta."end", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
-
((to_date(SUBSTR(ta."start",0,(INSTR(ta."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)
))/3600) AS avgH,
ta."name" AS name9
FROM view_task ta, view_instance vi
WHERE vi.definitionName = 'PROZESSDEFINITIONSNAME'
AND vi.id = ta.instanceId
AND ta."end" IS NOT NULL
GROUP BY ta."name"
)
WHERE name1 = name2
AND name1 = name3
AND name1 = name4
AND name1 = name5
AND name1 = name6
AND name1 = name7
AND name1 = name8
AND name1 = name9
)
```

Durchschnittliche, minimale und maximale Durchlaufszeit aller Aufgaben einer bestimmten Prozessdefinition (in h)

```
SELECT task."name" AS Aufgabe,
CAST((avg((((to_date(SUBSTR(task."end",0,(INSTR(task."end", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)  -
((to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24) ))/3600) AS
DECIMAL(10,1)) AS "Durschnittliche Durchlaufszeit",
CAST((MIN((((to_date(SUBSTR(task."end",0,(INSTR(task."end", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)  -
((to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24))/3600)) AS
DECIMAL(10,1)) "Minimale Durchlaufszeit",
CAST((MAX((((to_date(SUBSTR(task."end",0,(INSTR(task."end", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)  -
((to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24))/3600)) AS
DECIMAL(10,1))  AS "Maximale Durchlaufszeit"
FROM view_task task
INNER JOIN
```

```
view_instance inst ON task.instanceId=inst.id
WHERE task."end" IS NOT NULL
AND task."start" IS NOT NULL
AND (((to_date(SUBSTR(task."end",0,(INSTR(task."end", '.', -1)-1)), 'DD-MM-
YY HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24) -
((to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)) > 0
AND inst.definitionname = 'NAME DER PROZESSDEFINITIOn'
GROUP BY task."name"
```

Durchschnittliche, minimale und maximale Durchlaufzeit aller Aufgaben einer bestimmten Prozessdefinition (in h) je Monat

```
SELECT
CONCAT(CONCAT(SUBSTR(to_char(to_date(SUBSTR(task."start",0,(INSTR(task."star
t", '.', -1)-1)), 'DD-MM-YY HH24:MI:SS'), 'Month'),1,3),' '),EXTRACT(YEAR
FROM to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-
MM-YY HH24:MI:SS'))) AS Monat,
task."name" AS Aufgabe,
CAST((avg((((to_date(SUBSTR(task."end",0,(INSTR(task."end", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)  -
((to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24) ))/3600) AS
DECIMAL(10,1)) AS "Durschnittliche Durchlaufzeit",
CAST((MIN((((to_date(SUBSTR(task."end",0,(INSTR(task."end", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)  -
((to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24))/3600)) AS
DECIMAL(10,1)) "Minimale Durchlaufzeit",
CAST((MAX((((to_date(SUBSTR(task."end",0,(INSTR(task."end", '.', -1)-1)),
'DD-MM-YY HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)  -
((to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24))/3600)) AS
DECIMAL(10,1))  AS "Maximale Durchlaufzeit"
FROM view_task task
INNER JOIN
view_instance inst ON task.instanceId=inst.id
WHERE task."end" IS NOT NULL
AND task."start" IS NOT NULL
AND (((to_date(SUBSTR(task."end",0,(INSTR(task."end", '.', -1)-1)), 'DD-MM-
YY HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24) -
((to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY
HH24:MI:SS')  - DATE '1970-01-01' ) * 60 * 60 * 24)) > 0
AND inst.definitionname = 'PROZESSDEFINITIONSNAME'
GROUP BY
EXTRACT(YEAR FROM to_date(SUBSTR(task."start",0,(INSTR(task."start", '.',
-1)-1)), 'DD-MM-YY HH24:MI:SS')),
```

```
CONCAT(CONCAT(SUBSTR(to_char(to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY HH24:MI:SS'), 'Month'),1,3),' '),EXTRACT(YEAR FROM to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY HH24:MI:SS'))),
task."name",
EXTRACT(YEAR FROM to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY HH24:MI:SS')),
EXTRACT(MONTH FROM to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY HH24:MI:SS'))
ORDER BY
EXTRACT(YEAR FROM to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY HH24:MI:SS')),
EXTRACT(MONTH FROM to_date(SUBSTR(task."start",0,(INSTR(task."start", '.', -1)-1)), 'DD-MM-YY HH24:MI:SS'))
```