

Volume

4

HOSxP

Enterprise Hospital Information System



Report manual

HOSXP

Report manual

Chaiyaporn Suratemekul

Revision 1.0 January 2008
Revision 0.1 December 2004

คำนำ

หนังสือเล่มนี้จัดทำขึ้นเพื่อใช้ประกอบการอบรมผู้ดูแลระบบ HOSxP โดยเนื้อหาประกอบไปด้วยหัวข้อที่เกี่ยวข้องและจำเป็นต้องใช้ในการดูแลระบบ HOSxP ซึ่งข้าพเจ้าหวังเป็นอย่างยิ่งว่าผู้ที่ได้รับการอบรมจะสามารถนำความรู้ที่ได้รับ ในงานที่ตนเองรับผิดชอบได้อย่างมีประสิทธิภาพ

ชัยพร สุระเดมิย์กุล

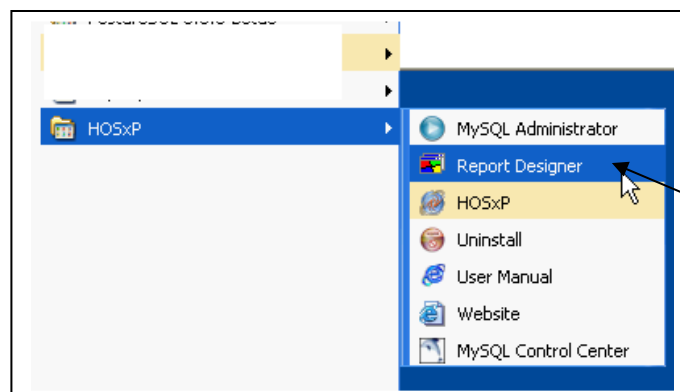
สารบัญ

การเข้าใช้งานระบบรายงาน	2
การใช้ Variable	16
การใช้งานระบบ Sub Report	29
การสร้าง Crosstab report	39
การสร้าง Custom Report	45
การ Download รายงานใหม่ๆ จาก Web Report Central	51

การเข้าใช้งานระบบรายงาน

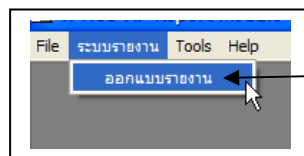
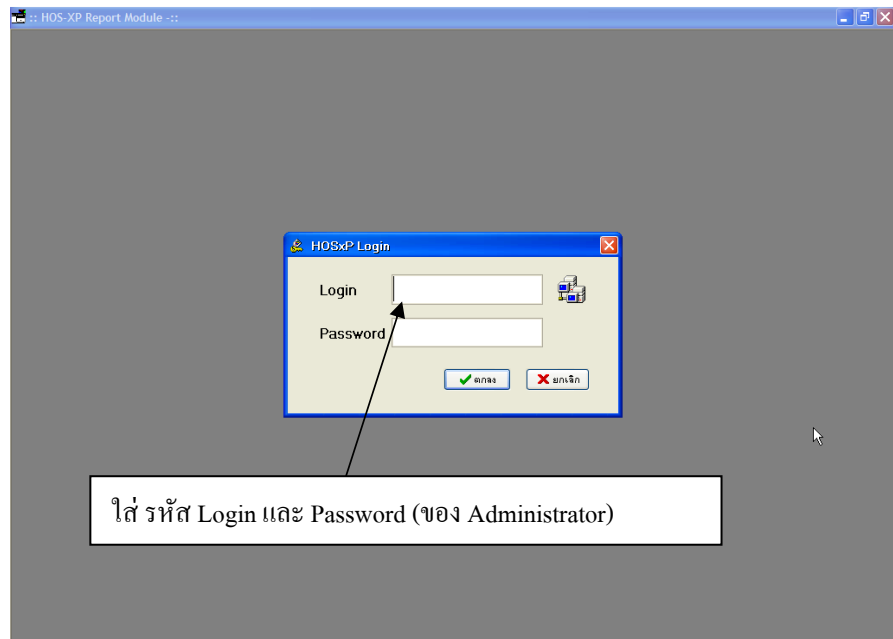
HOSxP มีระบบรายงานที่แยกออกมาโดยเฉพาะ ซึ่งแบบฟอร์มของรายงานต่างๆ จะถูกเก็บไว้ในฐานข้อมูล การแก้ไขรายงานใดๆ จะมีผลกับทุกเครื่องที่ใช้งาน รายงานจะแบ่งออกเป็น 2 กลุ่มหลักๆ คือ รายงานที่เป็น System Report และรายงานที่เป็น Custom Report ซึ่งผู้ดูแลสามารถเข้าไปแก้ไขรายงานในกลุ่มใดๆ ก็ได้

การเข้าใช้งานระบบรายงาน มีขั้นตอนดังนี้

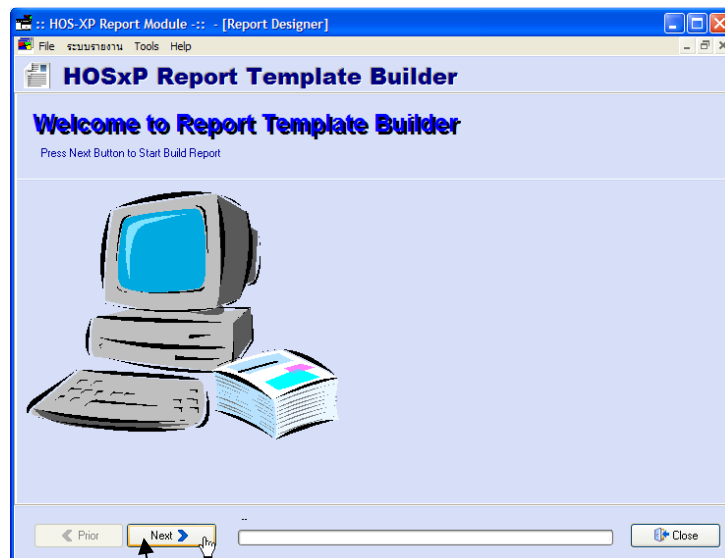


เลือก Report Designer
จาก Start Menu ->
Programs -> HOSxP

การติดตั้งโปรแกรม



เลือกคำสั่ง ออกแบบรายงาน จาก
เมนู ระบบรายงาน



กดปุ่ม Next เพื่อไปยังหน้าถัดไป

การติดตั้งโปรแกรม

Tab เลือกระหว่าง System Report กับ Custom Report

เลือกรายงานที่ต้องการแก้ไข

ชื่อรายงานที่กำลังเลือก

ชุดคำสั่งที่ใช้ในการดึงข้อมูลมาทำรายงาน

กด Next เพื่อไปยังหน้าถัดไป

หน้าจอชุดคำสั่ง SQL สามารถแก้ไขได้

Parameter ที่ตั้งไว้จะเป็นตัวสีน้ำเงิน

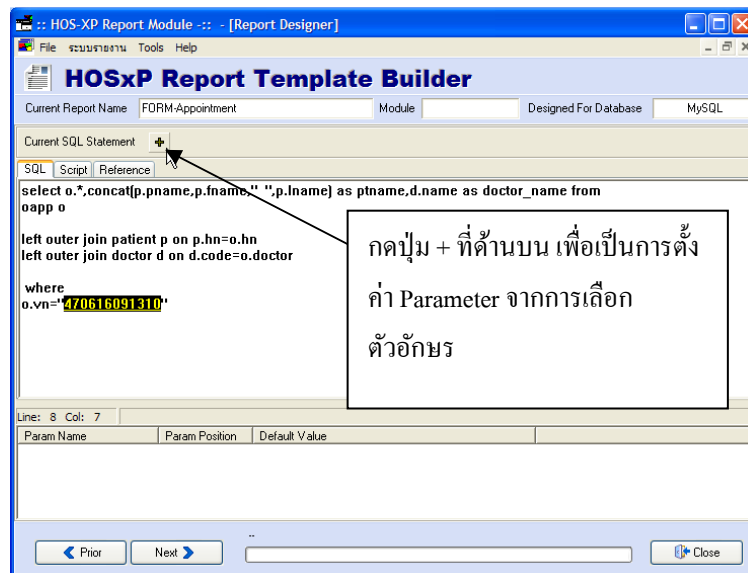
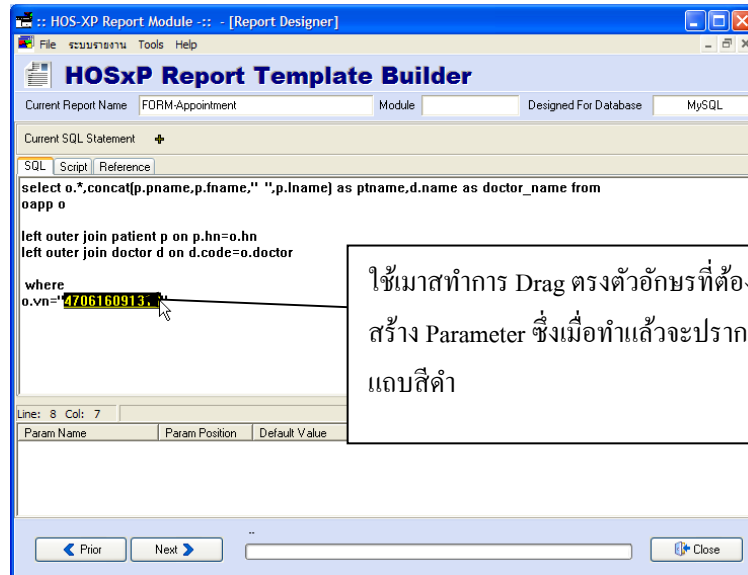
Tab ย่อยได้แก่ SQL, Script และ Reference

Parameter list แสดงช่วงของ Parameter ที่ได้ตั้งไว้

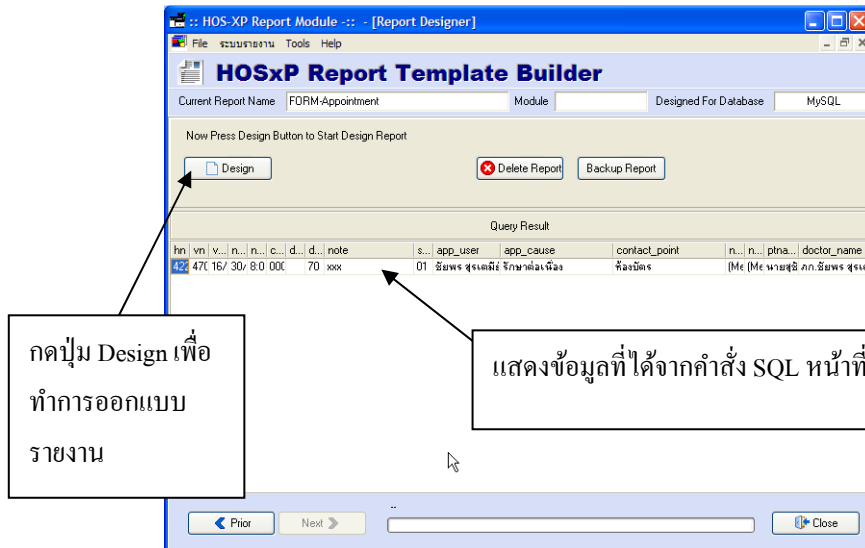
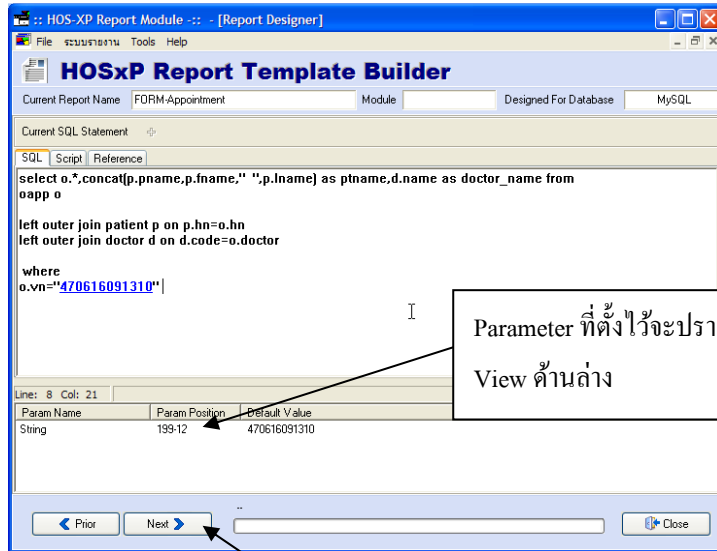
Param Name	Param Position	Default Value
String	199-12	470616091310

การติดตั้งโปรแกรม

ที่ช่อง SQL ผู้ออกแบบสามารถแก้ไขชุดคำสั่งเดิมได้ โดยทุกครั้งที่แก้ไข Parameter จะถูก Clear ออกเสมอ ดังนั้นก่อนจะแก้ไข จะต้องจำได้ว่า ชุดคำสั่งเดิมได้ตั้งค่า Parameter ไว้ตรงไหนบ้าง (ตัวที่ดึงไว้เป็น Parameter จะแสดงเป็นตัวสีน้ำเงิน) โดยเมื่อแก้ไขชุดคำสั่ง SQL แล้วสามารถสร้าง Parameter ได้ตามขั้นตอนดังนี้

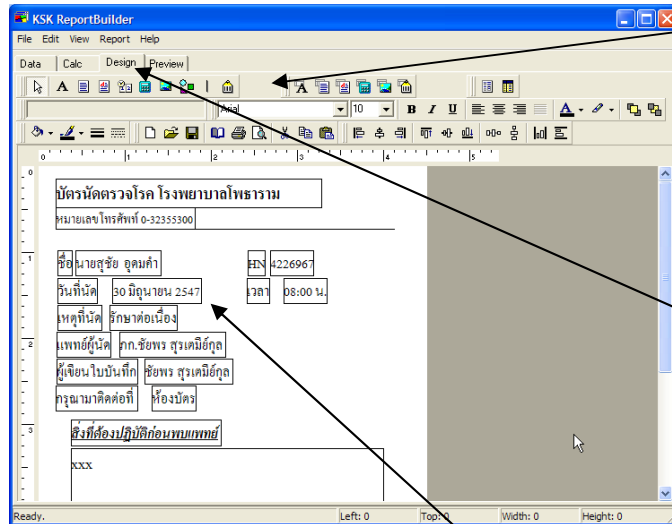


การติดตั้งโปรแกรม



การติดตั้งโปรแกรม

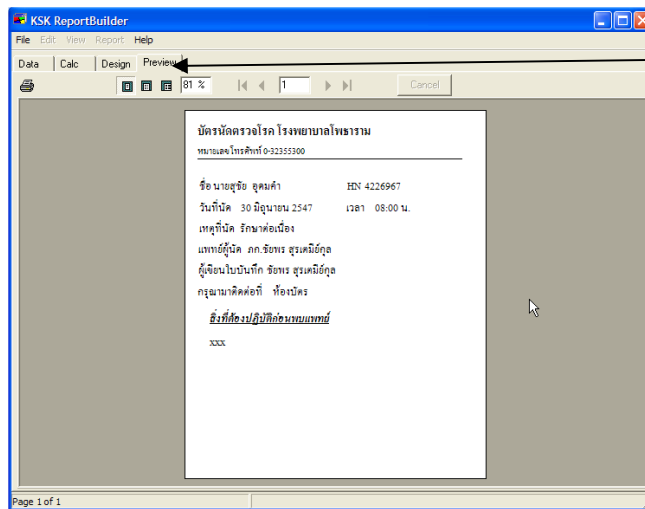
จะปรากฏหน้าจอออกแบบรายงานดังรูปด้านล่าง



Tool Bar ที่ใช้ในการ
เลือก Tool ทำรายงาน

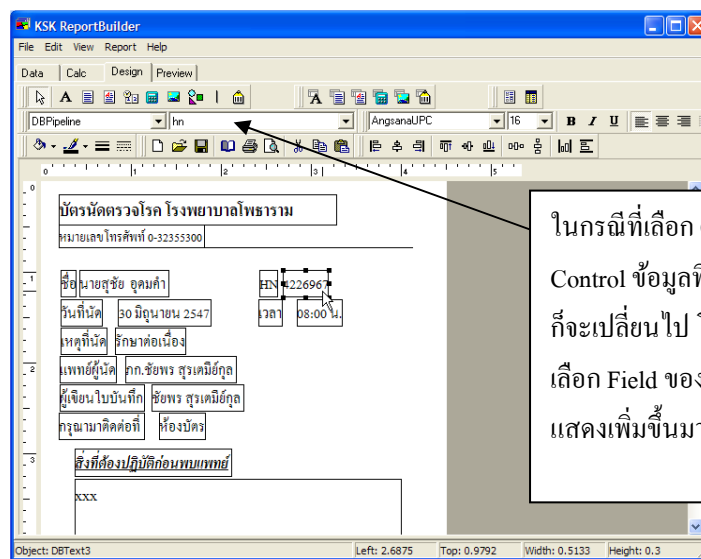
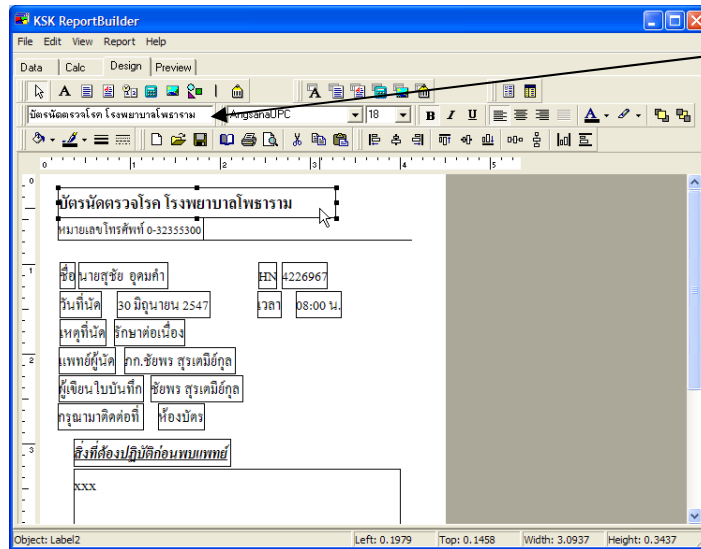
Tab เลือกแสดงรายงาน

หน้าจอแสดง ตัวอักษรต่างๆที่ได้จัด
ตำแหน่งไว้ สามารถปรับตำแหน่งได้

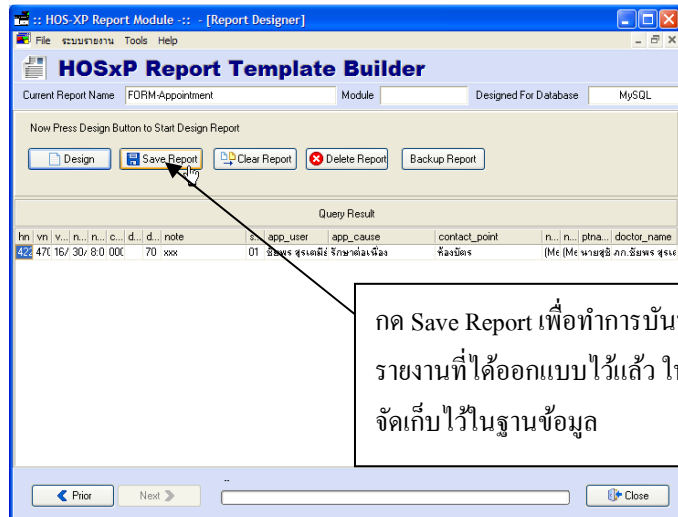
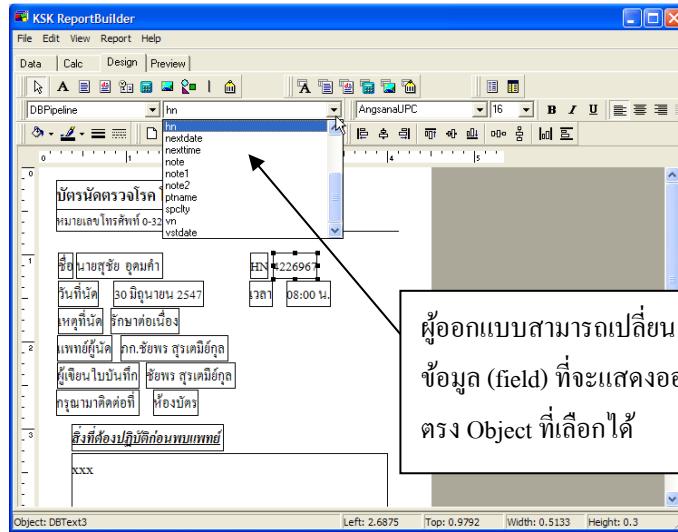


การกด Tab
Preview จะแสดง
ตัวอย่างรายงานที่
จะถูกพิมพ์ออกมา

การติดตั้งโปรแกรม



การติดตั้งโปรแกรม



การดึงโปรแกรม

ตัวอย่างคำสั่ง SQL ที่ใช้ในการทำรายงาน

จงแสดงรายชื่อผู้ป่วยที่มีชื่อแรกว่า สมชาย

```
SELECT *  
FROM patient  
WHERE fname = 'สมชาย'
```

จงแสดงรายชื่อผู้ป่วยที่มีนามสกุลว่า แข็งขัน แสดงเฉพาะ HN, ชื่อ, นามสกุล

```
SELECT hn, pname, fname, lname  
FROM patient  
WHERE lname = 'แข็งขัน'
```

จงแสดงรายชื่อผู้ป่วย ที่อยู่จังหวัด กรุงเทพมหานคร แสดงเฉพาะ HN, ชื่อ, นามสกุล

(เปิดตาราง thaiaddress ดูรหัสจังหวัด)

```
SELECT hn, fname, lname  
FROM patient  
WHERE chwpart='??'
```

แสดงรายชื่อผู้ป่วยที่มีชื่อ เหมือน สม

```
SELECT *  
FROM patient  
WHERE fname LIKE 'สม%'
```

```
SELECT COUNT(*)  
FROM patient  
WHERE fname LIKE 'สม%'
```

```
SELECT sex, pname, count(*)  
FROM patient  
WHERE fname LIKE 'สม%'  
GROUP BY sex, pname
```

```
SELECT *  
FROM patient
```

การรวม Character

ใช้ function CONCAT

```
SELECT hn, CONCAT(pname, fname, ' ', lname)  
FROM patient  
WHERE fname LIKE 'สม%'
```

การ ตั้งชื่อ column ใช้ AS

```
SELECT CONCAT(pname, fname, ' ', lname) as NAME  
FROM patient  
WHERE fname LIKE 'สม%'
```

การดึงข้อมูล

การระบุ Field ที่เป็น วันที่ ใช้รูปแบบ yyyy-mm-dd (ปี ค.ศ.) เช่น '2004-01-11' จงแสดงผู้ที่เกิดวันที่ 10 กุมภาพันธ์ 2510

```
SELECT *
from patient
where birthday = '1967-02-10'
```

จงแสดงผู้มารับบริการในวันที่ 10 สิงหาคม 2547 แสดงเฉพาะ HN, ชื่อ, เวลาที่มารับบริการ, ประเภทโรค, วันเกิด, ชื่อประเภทโรค ในช่วงเวลา 08.00 - 10.00 น. เฉพาะ ประเภท อายุรกรรม, จักษุ, สูติกรรม

```
SELECT ovst.hn,ovst.vsttime,ovst.spclty,
concat(patient.fname,' ',patient.lname) as patient_name,
patient.birthday , spclty.name
FROM ovst
LEFT OUTER JOIN patient on patient.hn=ovst.hn
LEFT OUTER JOIN spclty on spclty.spclty=ovst.spclty
WHERE ovst.vstdate = '2004-08-10'
AND ovst.vsttime >= '08:00:00'
AND ovst.vsttime <= '10:00:00'
AND ovst.spclty IN ( '01' , '07' , '03' )
ORDER BY spclty.name
```

จงหาว่าในวันที่ 20 กรกฎาคม 2547 มีผู้มารับบริการในประเภทโรคต่างๆ กี่ครั้ง

การตัดข้อมูลซ้ำกันออกใช้ DISTINCT เช่น count(distinct hn)

จงหาว่า ในเดือน สิงหาคม มีผู้ป่วยมารับบริการ กี่คน

จงหาว่า ในเดือน สิงหาคม มีผู้ป่วยมารับบริการ กี่ครั้ง

```
SELECT count(distinct hn) from ovst where
vstdate between '2004-08-01' and '2004-08-31'
```

```
SELECT count(hn) from ovst where
vstdate between '2004-08-01' and '2004-08-31'
```

จงหาว่า ในเดือน สิงหาคม มีผู้ป่วยมารับบริการใหม่ในปี กี่คน

จงหาว่า ในเดือน สิงหาคม มีผู้ป่วยมารับบริการใหม่ในเดือน กี่คน

```
SELECT count(distinct hn) from vn_stat where
vstdate between '2004-08-01' and '2004-08-31'
AND count_in_year = 0
```

จงหาว่า ในเดือน สิงหาคม มีผู้ป่วย Admit กี่คน (ipt)

จงหาว่า ในเดือน สิงหาคม มีผู้ป่วย Admit กี่ครั้ง (ipt)

จงหาว่าในเดือนสิงหาคม 2547 มีผู้ป่วยเป็นโรคต่างๆ อะไรบ้าง กี่ครั้ง (แสดง pdx,จำนวน)

เรียงลำดับ แสดง 20 รายการแรก (จากมากไปน้อย) (vn_stat) ORDER BY ... DESC

การดึงข้อมูล

```
SELECT pdx,count(*) as pdx_count
FROM vn_stat
WHERE vstdate between '2004-08-01' and '2004-08-31'
GROUP BY pdx
ORDER BY pdx_count DESC
LIMIT 20
```

แสดงชื่อโรคด้วย (icd10)

```
SELECT v.pdx,i.name,count(v.pdx) as pdx_count
FROM vn_stat v
LEFT OUTER JOIN icd10 i on i.code=v.pdx
WHERE v.vstdate between '2004-08-01' and '2004-08-31'
GROUP BY v.pdx,i.name
ORDER BY pdx_count DESC
LIMIT 20
```

การแสดงที่อยู่ โดยการทำ multiple join

```
SELECT p.hn,p.fname,p.lname,
t1.name as chw_name, t2.name as amp_name,
t3.name as tmb_name
FROM patient p
LEFT OUTER JOIN thaiaddress t1 on t1.chwpart=p.chwpart
and t1.codetype='1'
LEFT OUTER JOIN thaiaddress t2 on t2.chwpart=p.chwpart
and t2.ampart=p.ampart and t2.codetype='2'
LEFT OUTER JOIN thaiaddress t3 on t3.chwpart=p.chwpart
and t3.ampart=p.ampart
and t3.tmbpart=p.tmbpart and t3.codetype='3'
WHERE p.fname LIKE 'สมช%'
```

จงหาจำนวนของผู้ป่วยที่อยู่ในตำบลต่างๆ ใน อ.โพธาราม แยกเป็น รหัสตำบล,ชื่อตำบล , จำนวนผู้ป่วยในตำบล

```
SELECT p.tmbpart,count(p.hn) as patient_count,
t.name as tmb_name
FROM patient p

LEFT OUTER JOIN thaiaddress t on t.chwpart=p.chwpart
AND t.ampart=p.ampart AND t.tmbpart=p.tmbpart
AND t.codetype='3'

WHERE p.chwpart='70' AND p.ampart='07'
GROUP BY p.tmbpart,t.name
```

จงหาค่าใช้จ่ายทั้งหมดของผู้ป่วยนอกที่มาใช้บริการในเดือน สิงหาคม (vn_stat.income)

```
select sum(income) from vn_stat where vstdate between '2004-08-01' and '2004-08-31'
```

การดึงข้อมูล

จงหาว่าโรคอะไรที่มีค่าใช้จ่ายรวมแล้วมากที่สุด ของเดือนสิงหาคม
(vn_stat.main_pdx, vn_stat.income)

```
SELECT main_pdx,sum(income) as sum_income
FROM vn_stat
WHERE vstdate between '2004-08-01' and '2004-08-31'
GROUP BY main_pdx
ORDER BY sum_income DESC
LIMIT 20
```

จงหาว่าแพทย์ท่านใดตรวจคนไข้มากที่สุดในเดือน สิงหาคม
(vn_stat.dx_doctor)

```
SELECT dx_doctor,count(dx_doctor) as count_dx
FROM vn_stat
WHERE vstdate between '2004-08-01' and '2004-08-31'
GROUP BY dx_doctor
ORDER BY count_dx DESC
LIMIT 20
```

และแพทย์ท่านนี้ สั่งจ่ายยา/เวชภัณฑ์ ในเดือนสิงหาคม มีมูลค่าเท่าไร
select sum(inc12) from vn_stat where dx_doctor='19' and
vstdate between '2004-08-01' and '2004-08-31'

จงหาว่าโรคใดที่ผู้ป่วยกลับมารักษาภายใน 7 วัน มากที่สุดในเดือนสิงหาคม (vn_stat.lastvisit)

```
SELECT main_pdx,count(pdx) as pdx_count
FROM vn_stat
WHERE lastvisit <= 7 AND vstdate between '2004-08-01' and  
'2004-08-31'

GROUP BY main_pdx
ORDER BY pdx_count DESC
LIMIT 20
```

ตาราง opitemrece รหัสเวชภัณฑ์ icode รายละเอียดอยู่ในตาราง s_drugitems หรือ drugitems จงหา
ว่ายาตัวใดถูกสั่งจ่ายมากที่สุดในเดือนสิงหาคม (opitemrece.icode)

```
select o.icode,s.name,sum(o.qty) as sum_qty
from opitemrece o
left outer join s_drugitems s on s.icode=o.icode
where o.vstdate between '2004-08-01' and '2004-08-31'
group by o.icode ,s.name
order by sum_qty desc
```

การดึงโปรแกรม

จงหาว่ายาตัวใดมีมูลค่าการใช้มากที่สุดในเดือนสิงหาคม (opitemrece.icode)

```
select o.icode,s.name,sum(o.qty*o.unitprice) as sum_price
from opitemrece o
left outer join s_drugitems s on s.icode=o.icode
where o.vstdate between '2004-08-01' and '2004-08-31'
and o.icode like '1%'
group by o.icode ,s.name
order by sum_price desc
```

จงแสดงรายการยาและจำนวน ที่มีการใช้ในผู้ป่วยที่เป็นโรคเบาหวาน ในเดือนสิงหาคม

```
select o.icode,d.name,sum(o.qty) as pay_qty
from vn_stat v, opitemrece o, drugitems d
where v.vstdate between '2004-08-01' and '2004-08-31'
and v.main_pdx like 'E1%' and v.vn = o.vn
and o.icode=d.icode
group by o.icode,d.name
order by pay_qty desc
```

ระยะเวลา service_time

เวลาที่ลงทะเบียน = service3

เวลาพิมพ์ใบเสร็จ = service7

total time = service7 - service3

ตรวจสอบข้อมูล แสดง

```
select count(*),(sum((service7-service3))/count(*)) as
total_time from service_time
where vstdate between '2004-09-01' and '2004-09-30'
and service3 is not null and service6 is not null
and service6>service3
and (service7 - service3) < 18000
and service3 > '09:30:00'
order by vn desc
```

จงหาจำนวนรายการเวชภัณฑ์ยา ที่ถูกจ่ายให้ผู้ป่วยใน ในเดือนสิงหาคม 2547

```
select an,rxdate, count(icode) as icode_count
from opitemrece
where vstdate between '2004-09-01' and '2004-09-30'
and an is not null and an <>''
and icode like '1%'
group by an,rxdate
```

การติดตั้งโปรแกรม

การใช้ Variable

ในการออกแบบรายงาน ผู้ออกแบบสามารถใช้ Variable component ใน Report designer เพื่อช่วยดึงข้อมูลมาใส่ในรายงานได้โดยไม่ต้องแก้ไขคำสั่งหลักของรายงาน และในบางกรณี Variable จะช่วยแก้ปัญหาที่วิธีการปกติทำไม่ได้หรือทำได้แต่ใช้เวลานาน

ยกตัวอย่างเช่น หากต้องทำรายงานสรุปยอดการใช้ยาทุกตัวในปี แยกจำนวนใบสั่ง และมูลค่า โดยต้องการแสดงผลลัพธ์ดังนี้

รหัส | ชื่อ | จำนวนใบสั่ง | มูลค่า

คำสั่งเริ่มต้นอาจจะเป็นดังนี้

```
SELECT i.icode,i.name,i.strength,i.units,  
count(DISTINCT o.vn) AS prescribe_count,  
count(o.icode) AS item_count, sum(o.qty) AS  
sum_qty,sum(o.sum_price) AS tot_price
```

```
FROM drugitems i
```

```
LEFT OUTER JOIN opitemrece o ON o.icode = i.icode
```

```
WHERE o.rxdate BETWEEN '2005-01-01'  
AND '2005-12-31'
```

```
GROUP BY i.icode,i.name,i.strength,i.units
```

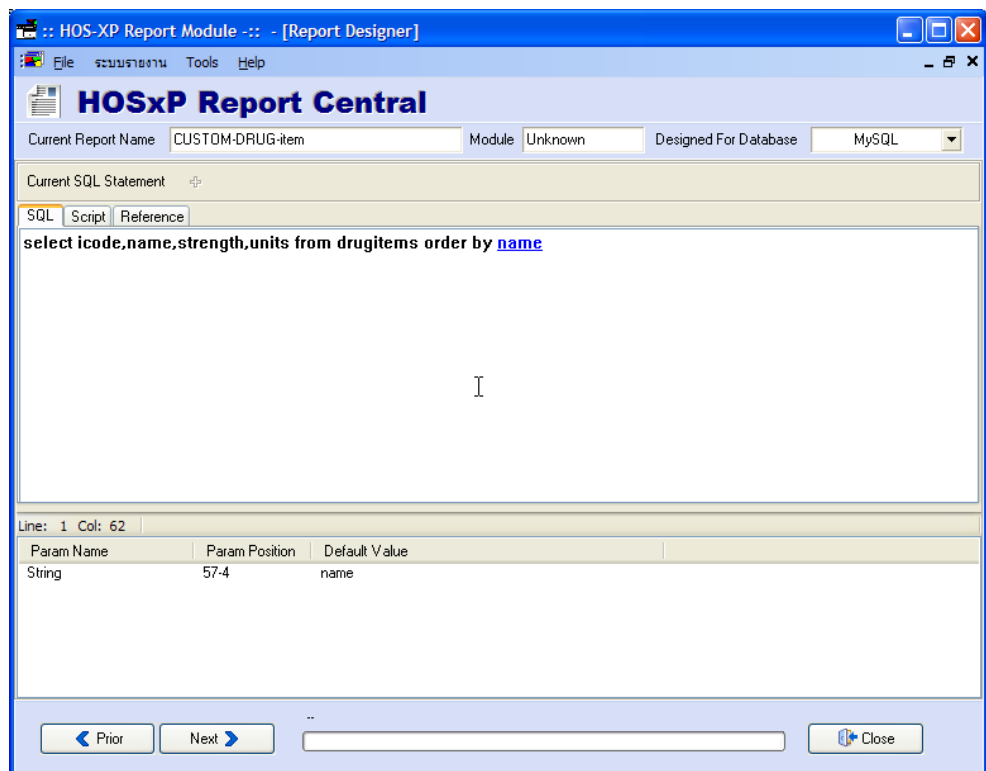
การใช้ VARIABLE

แต่ปัญหาก็คือหากข้อมูลที่นำมาใช้ทำรายงานมีจำนวนเยอะมาก ระยะเวลาที่ใช้ทำรายงานก็จะนานมากเช่นเดียวกัน

คราวนี้มาคิดว่า Variable จะช่วยแก้ปัญหานี้ได้อย่างไร เริ่มต้นจากเปลี่ยนคำสั่งให้แสดงแค่ข้อมูลยาทุกตัวก่อน แล้วค่อยไปคำนวณจำนวนในระบบรายงานครับ โดยการเปลี่ยนคำสั่งที่ใช้ทำรายงานเป็นแบบ simple แบบนี้

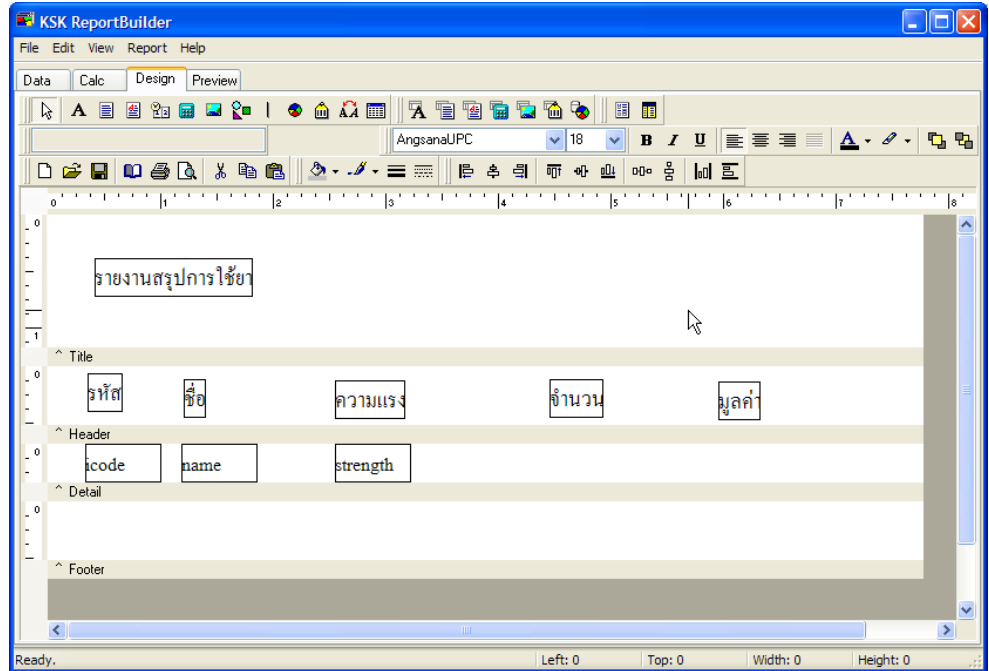
```
SELECT icode,name,strength,units FROM drugitems ORDER BY name
```

จากนั้นเอาไปเป็นคำสั่งสำหรับทำรายงาน ดังรูป

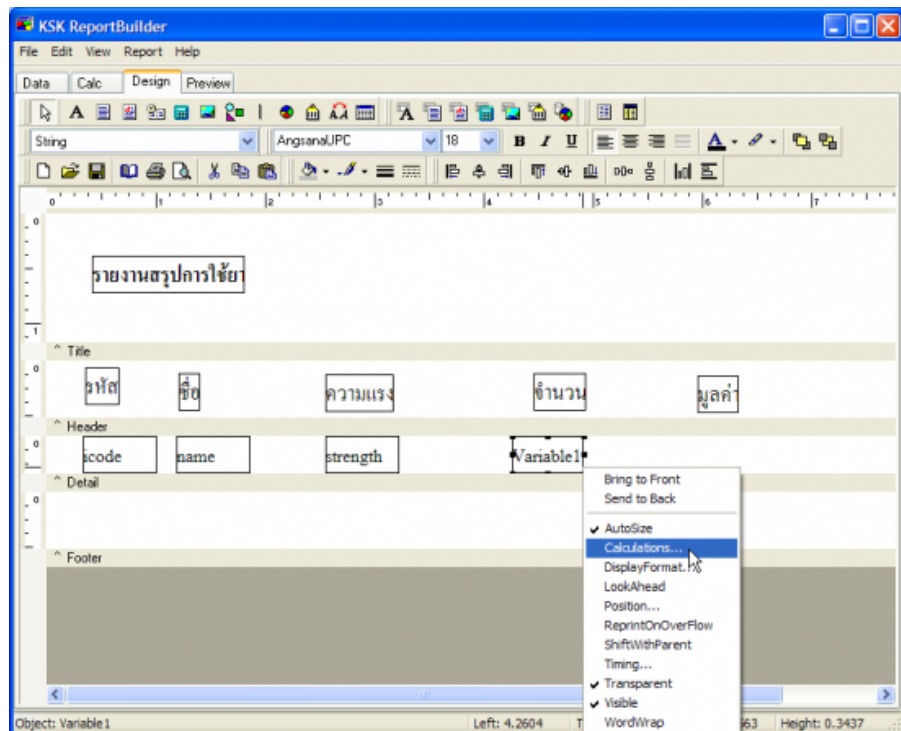


การใช้ VARIABLE

ออกแบบหน้าจอให้เรียบร้อยดังรูป แต่ปัญหาอยู่ที่การเอาจำนวน และ มูลค่าของยาแต่ละตัว มาใส่ในรายงาน เนื่องจากคำสั่ง SQL ที่ใช้ ไม่ได้ดึงข้อมูลนี้มาเลย



เราจะใช้ Variable เข้ามาช่วยครับ อันดับแรก วาง Variable ลงไปใน Detail Band ดังรูป จากนั้นคลิกขวา เลือก Calculations



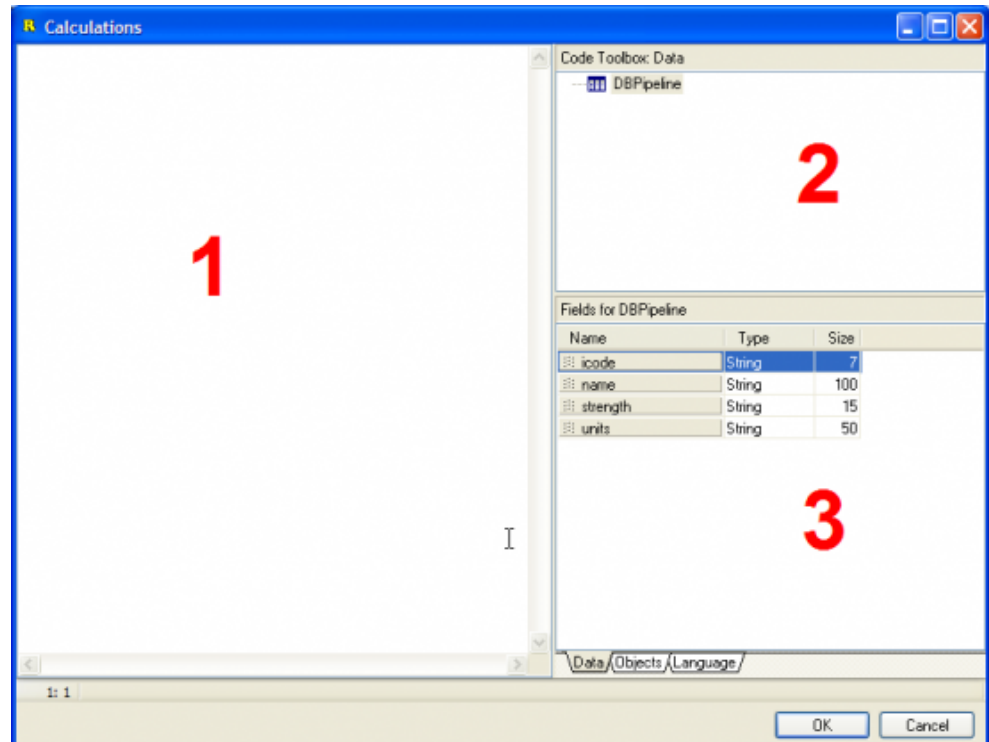
การใช้ VARIABLE

จะปรากฏหน้าต่าง Calculations ดังรูป

1 คือช่องสำหรับเขียน Pascal cod

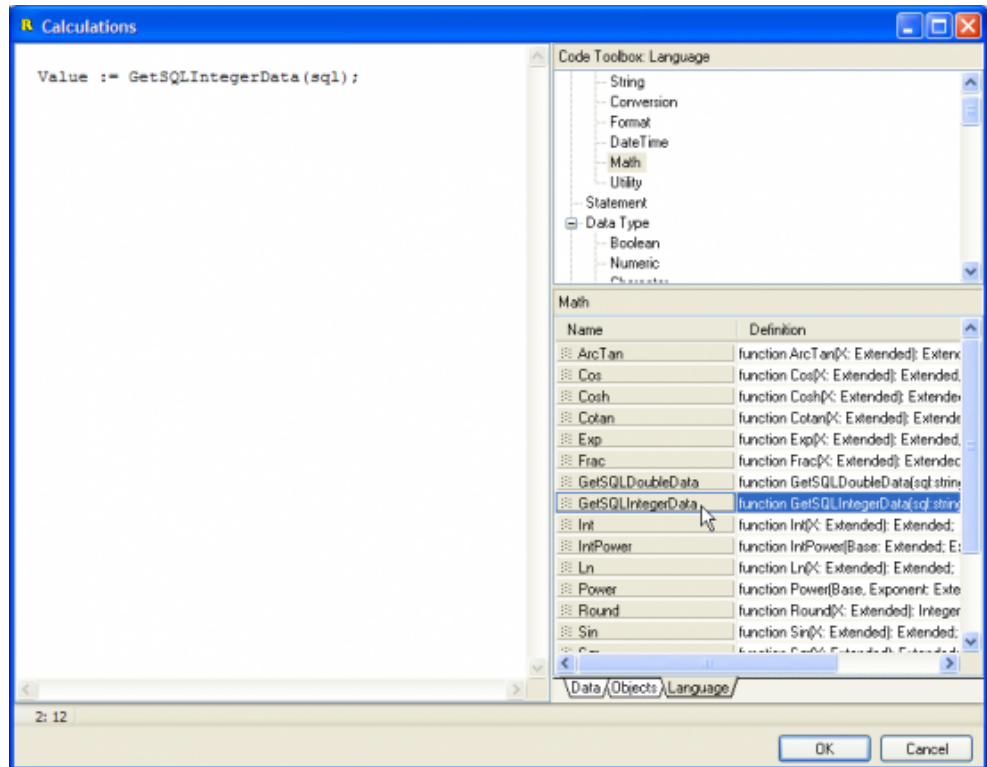
2 คือช่องสำหรับเลือกข้อมูลจะเปลี่ยนตาม tab ของ 3

3 คือชนิดของข้อมูลที่จะเลือกมาใช้ มี 3 แบบคือ Data , Objects และ Language



การใช้ VARIABLE

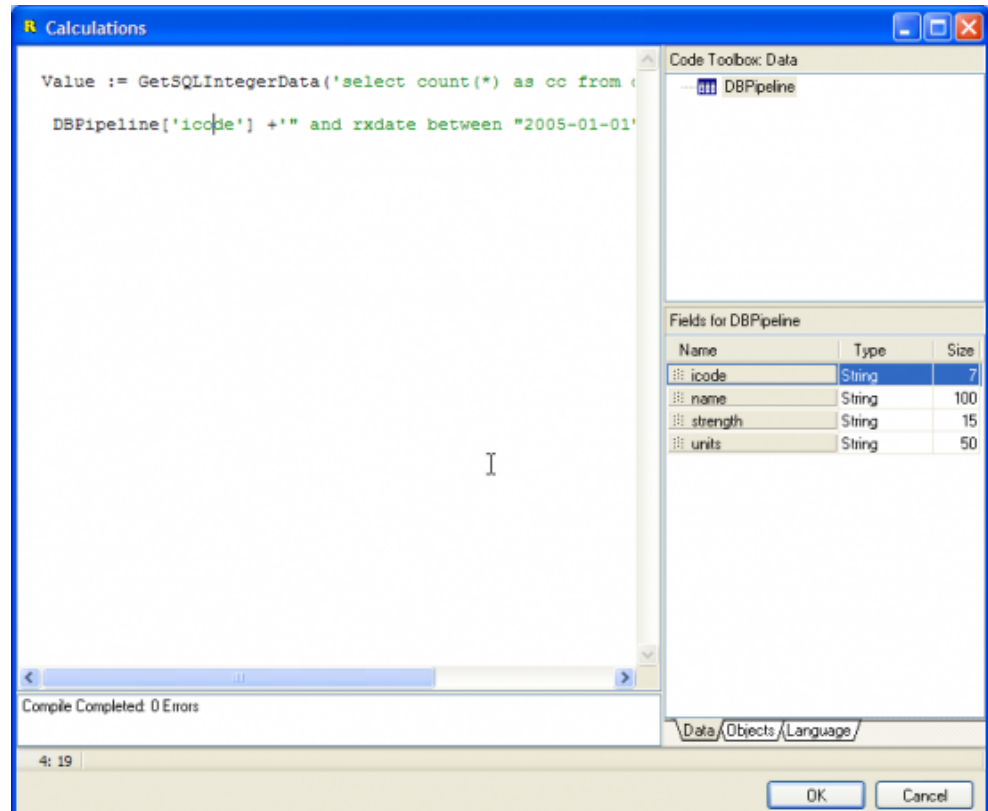
เลือก tab Language แล้วที่ช่อง 2 เลือก Math จากนั้นลากแถวที่เขียนว่า GetSQLIntegerData จากช่อง 3 มาไว้ในช่อง 1



การใช้ VARIABLE

จากนั้นแก้ไข pascal code ใหม่ให้เป็นอย่างนี้

```
Value := GetSQLIntegerData('select count(*) as cc from opitemrece where icode="' +  
DBPipeline['icode'] +'" and rxdate between  
"2005-01-01" and "2005-12-31" ');
```



การใช้ VARIABLE

จากนั้นกดปุ่ม Ok ก็จะกลับมาที่ Designer ให้ลองคลิกที่ Tab preview เพื่อดูตัวอย่าง ก็จะ
ได้ผลดังรูป

รหัส	ชื่อ	ความแรง	จำนวน	มูลค่า
1470006	0.45 % so	0.45 %	11	
1470019	10%Urea+	.. %	54	
1440106	ACETHY	5 g.	519	
1143001	ACTIFED	6 mg./ml.	143	
1000007	ACTIFED	60 mg.	1903	
1440403	ACYCLO	1 g.	57	
1470036	ACYCLO	200 mg.	26	
1000010	ADRENA	1 mg./ml.	16	

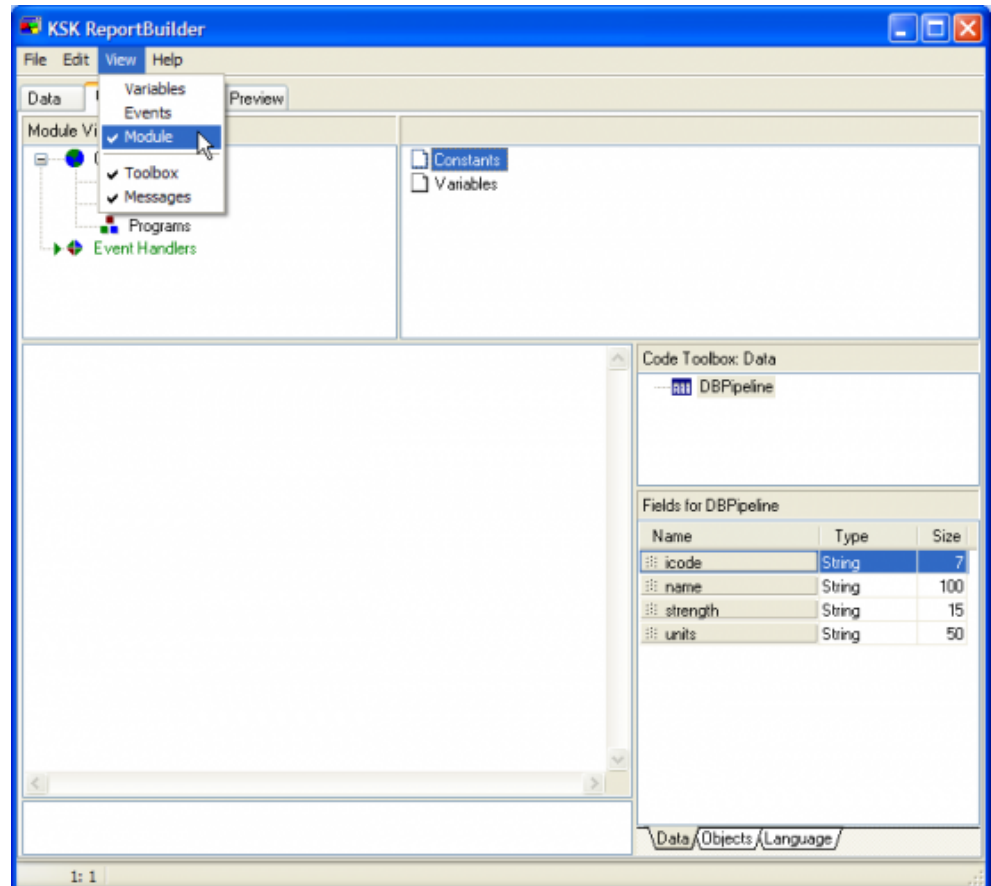
ปกติ Variable จะนำค่าที่ได้จากตัวแปรที่ชื่อ Value ไปแสดงผล Value จะเป็นตัวแปรชนิด Variant หมายความว่า จะกำหนด datatype ให้เป็นชนิดไหน ก็ได้ ส่วน Function GetSQLIntegerData เป็น function ที่นำคำสั่ง sql ส่งไปให้ database server ประมวลผล และส่งผลที่ได้ field แรก จาก record แรกมา (จากคำสั่งจะได้ผลลัพธ์ 1 record และ 1 field)

ส่วน DBPipeLine['icode'] ใน function GetSQLIntegerData หากระบุ DBPipeLine['icode'] หมายความว่า นำข้อมูลใน field icode ที่อยู่ใน dataset มาใช้ (dataset ที่ได้ มาจากคำสั่ง select * from drugitems) ซึ่งก็ขึ้นอยู่กับว่า cursor ปัจจุบันอยู่ตรงตำแหน่งไหน ปกติตัว report engine จะทำการวนจาก record แรก ไปยัง record สุดท้าย ดังนั้นค่าของ DBPipeLine['icode'] ก็จะเปลี่ยนไปเรื่อยๆ (จาก record แรก ไปยัง record สุดท้าย) และควรวาง Variable ให้ถูกตำแหน่งด้วย จากตัวอย่างวางไว้ใน Detail Band

การใช้ VARIABLE

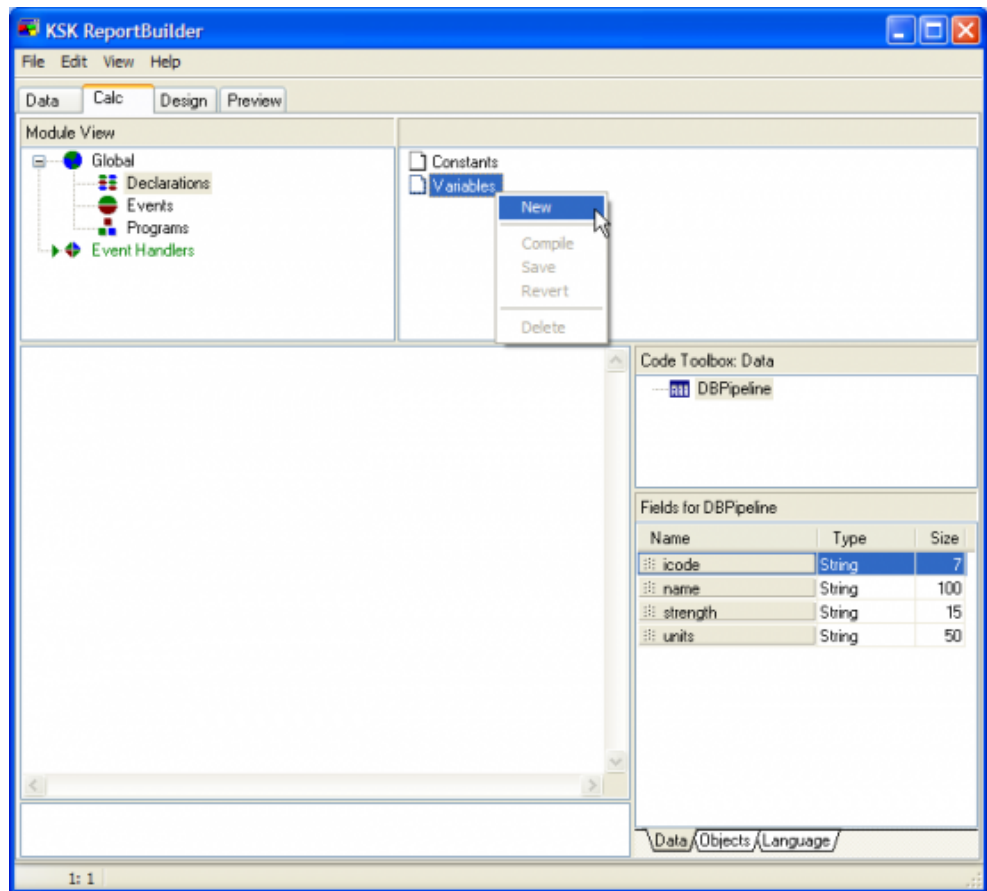
ในระบบรายงานนอกจากจะเขียน variable ให้ดึงข้อมูลจากฐานข้อมูลมาใช้งานได้แล้ว เรายังสามารถกำหนดให้ระบบรายงานทำการสอบถามผู้ใช้ให้ใส่วันที่และนำข้อมูลวันที่ๆ ผู้ใช้ใส่เข้าไปช่วยประมวลผลในการทำรายงานได้ด้วย

อันดับแรก คลิกที่แถบ Calc แล้วเลือกเมนู View -> Module



การใช้ VARIABLE

จากนั้นคลิกขวาที่ Variables แล้วเลือก New



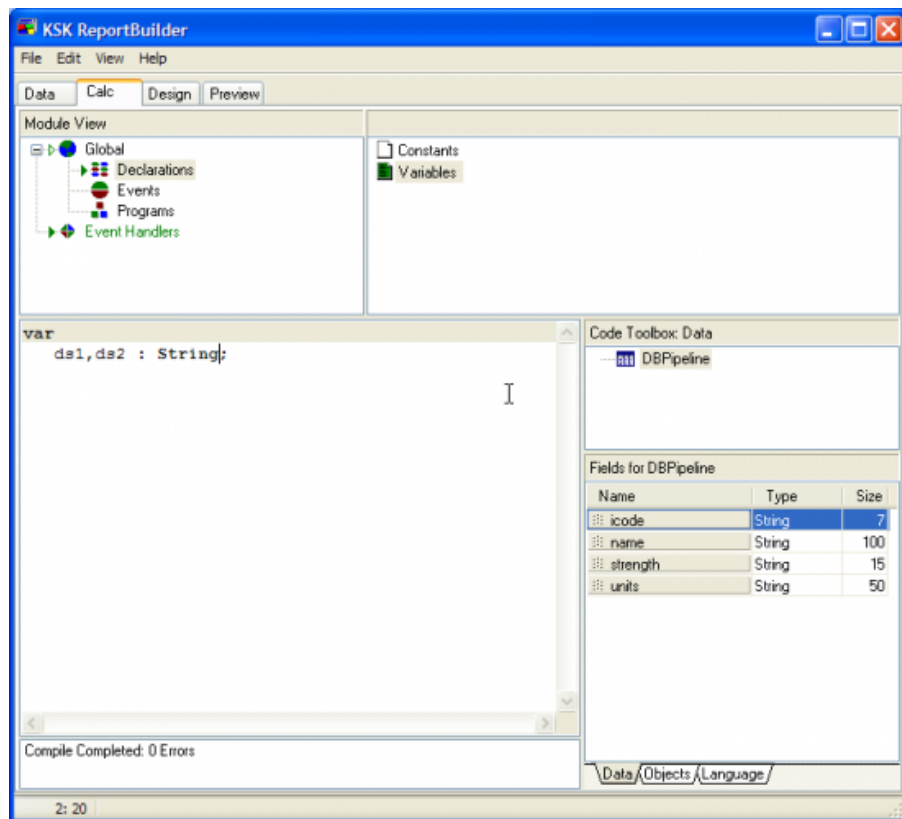
เขียน Code ที่ช่องด้านล่าง ดังนี้

```
var
```

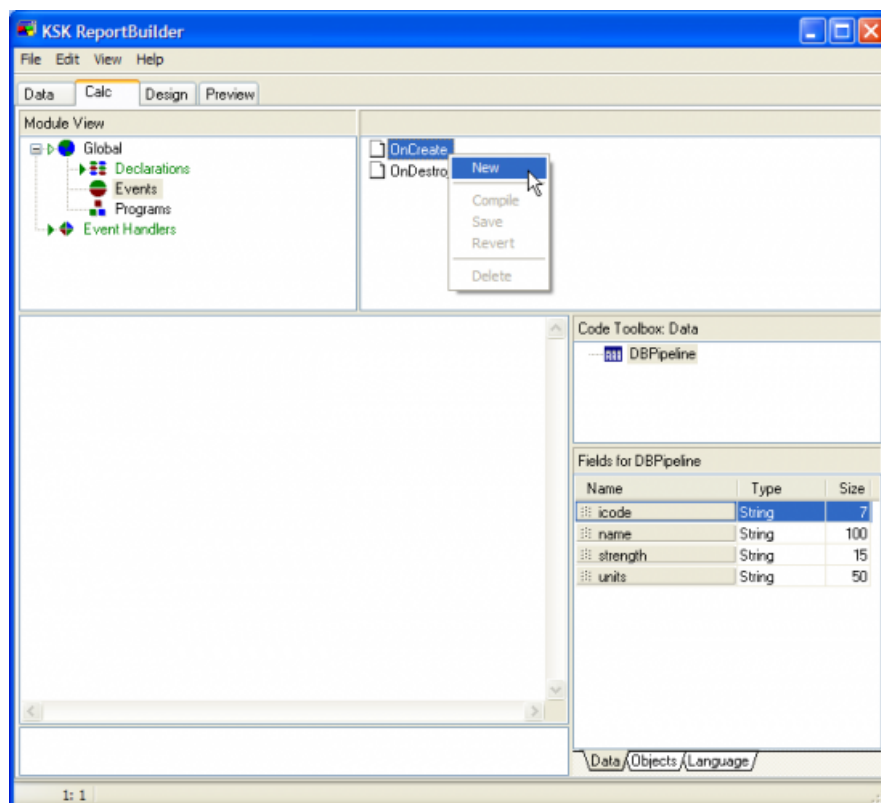
```
ds1,ds2 : String;
```

เพื่อประกาศตัวแปรขึ้นมา 2 ตัว ชื่อ ds1 กับ ds2 เป็นประเภท String (ตัวอักษร) เพื่อที่จะนำมาใช้เก็บวันที่ๆ จะให้ผู้ใช้เลือกช่วงวันที่ของรายงาน

การใช้ VARIABLE



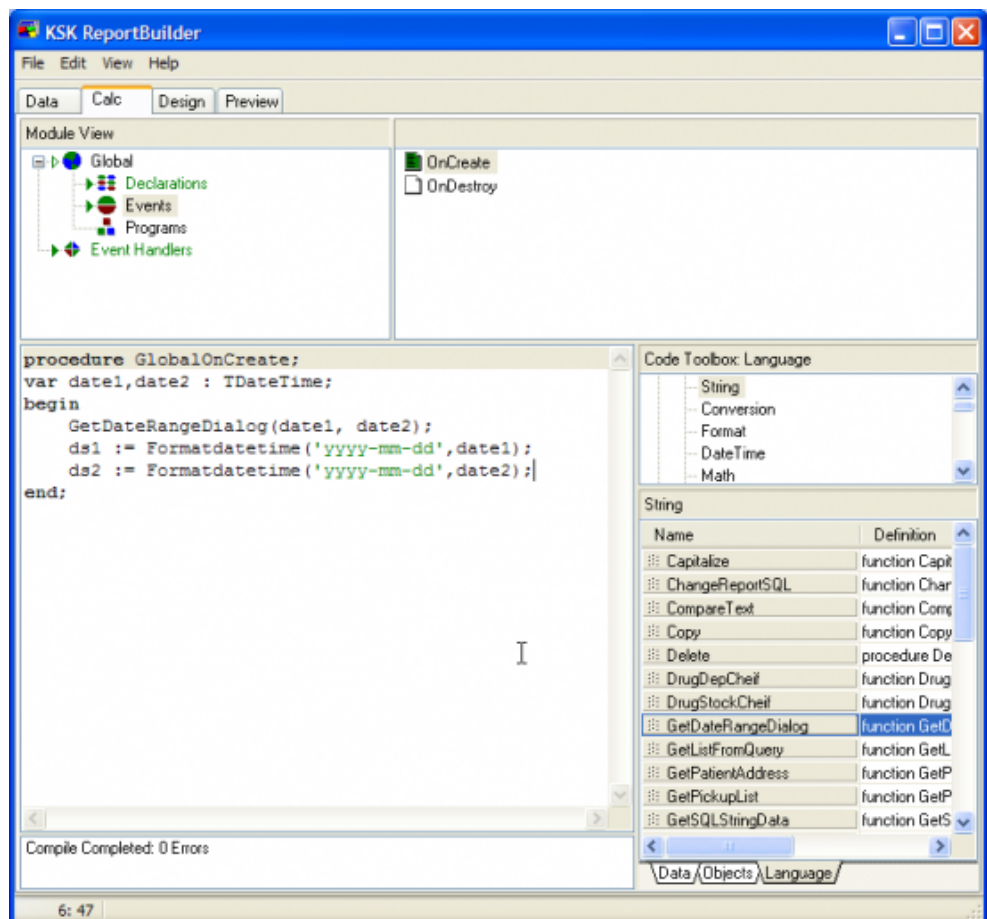
จากนั้นในช่อง Module View ให้คลิกที่ Events แล้วคลิกขวา ตรง onCreate แล้วเลือก New



การใช้ VARIABLE

จากนั้นเขียน Code ในช่องด้านล่าง ดังนี้

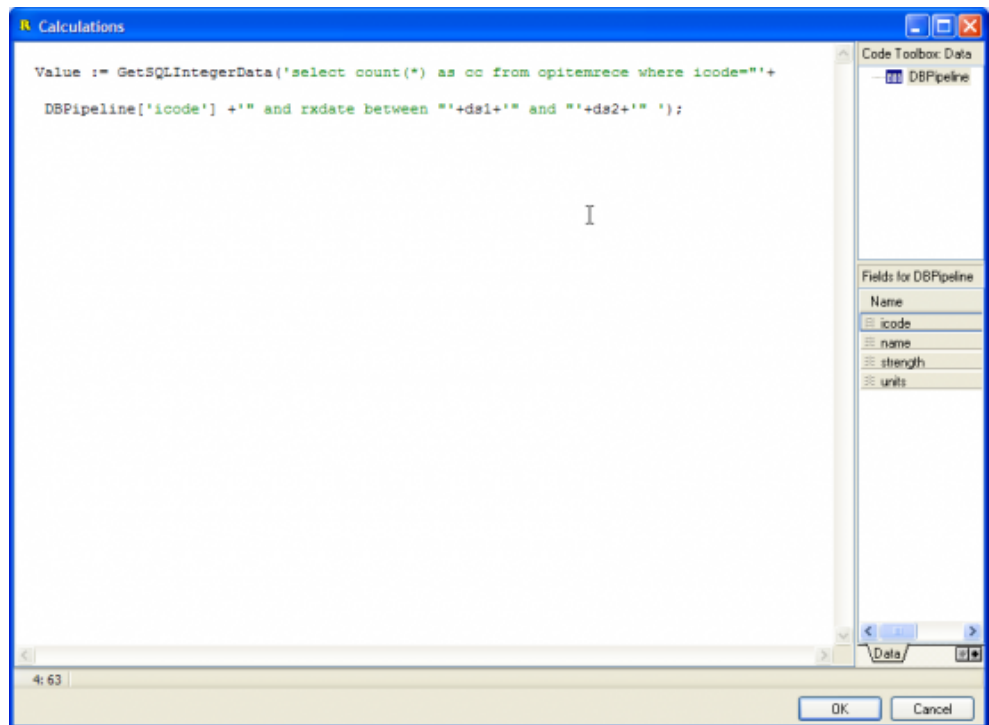
```
Procedure GlobalOnCreate;  
  
var datel,date2 : TDateTime;  
  
begin  
  
GetDateRangeDialog(datel, date2);  
  
ds1 := Formatdatetime('yyyy-mm-dd',datel);  
  
ds2 := Formatdatetime('yyyy-mm-dd',date2);  
  
end;
```



การใช้ VARIABLE

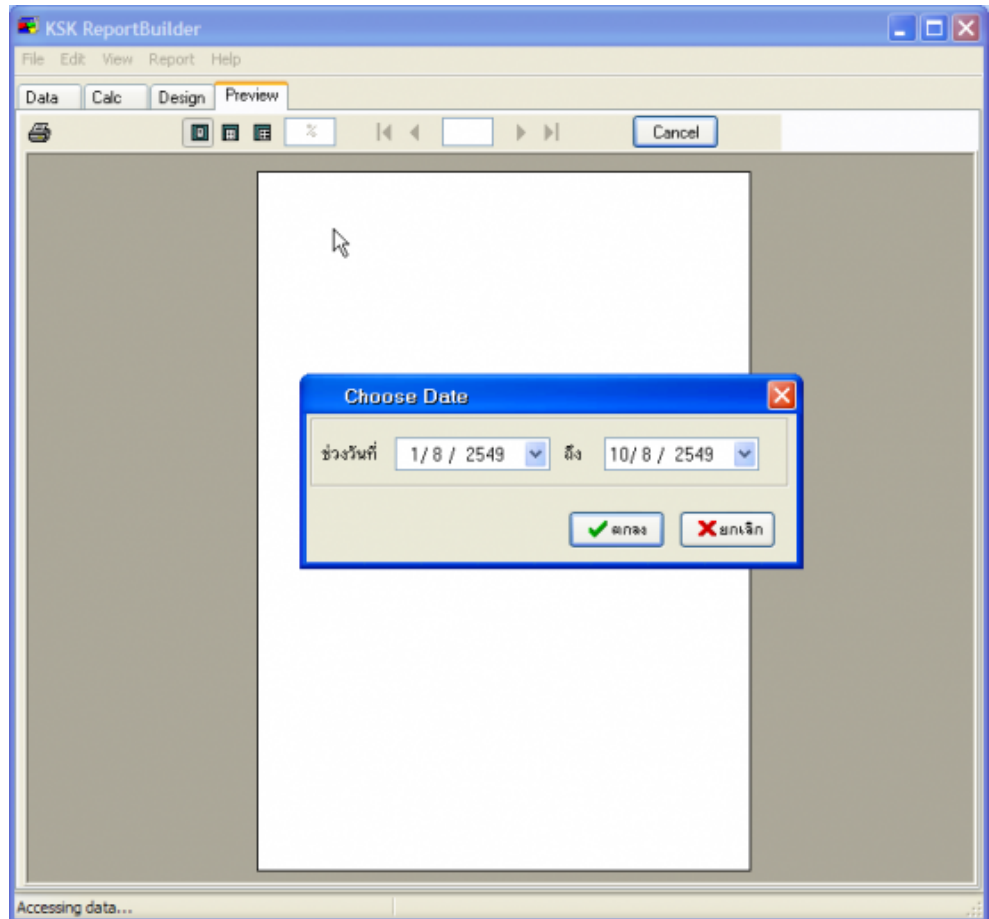
จากนั้นกลับมาที่ Tab Design แล้วคลิกขวาตรง Variable เพื่อแก้ไขคำสั่งที่ดึงข้อมูลมาแสดง โดยแก้ไขคำสั่งให้เป็น

```
Value := GetSQLIntegerData('select count(*) as cc from opitemrece where icode="'+  
DBPipeline['icode'] +' " and rxdate between  
"' +ds1+' " and "' +ds2+' " ');
```



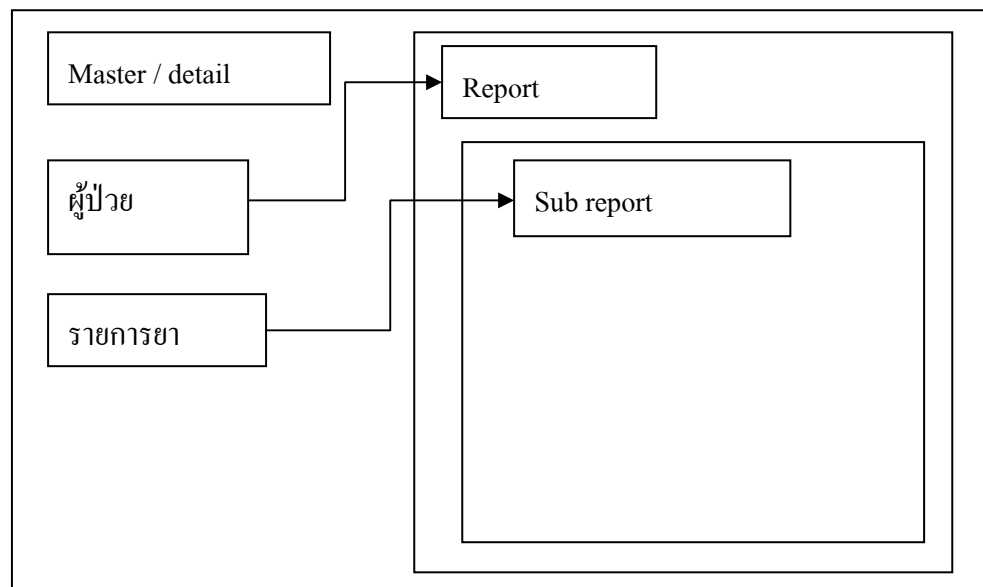
การใช้ VARIABLE

เสร็จแล้วก็ลองคลิกที่ tab Preview เพื่อทดสอบรายงาน จะปรากฏหน้าต่างแสดงให้ผู้ใช้เลือกวันที่ และเมื่อผู้ใช้กดปุ่มตกลง ระบบก็จะนำข้อมูลตามช่วงวันที่ๆ เลือกเอาไว้มาทำรายงาน



การใช้งานระบบ Sub Report

ระบบ Sub report ถูกออกแบบมาเพื่อช่วยในการสร้างรายงานที่มีลักษณะเป็น Master – detail โดยตัวอย่างรายงานจะมีลักษณะดังรูป



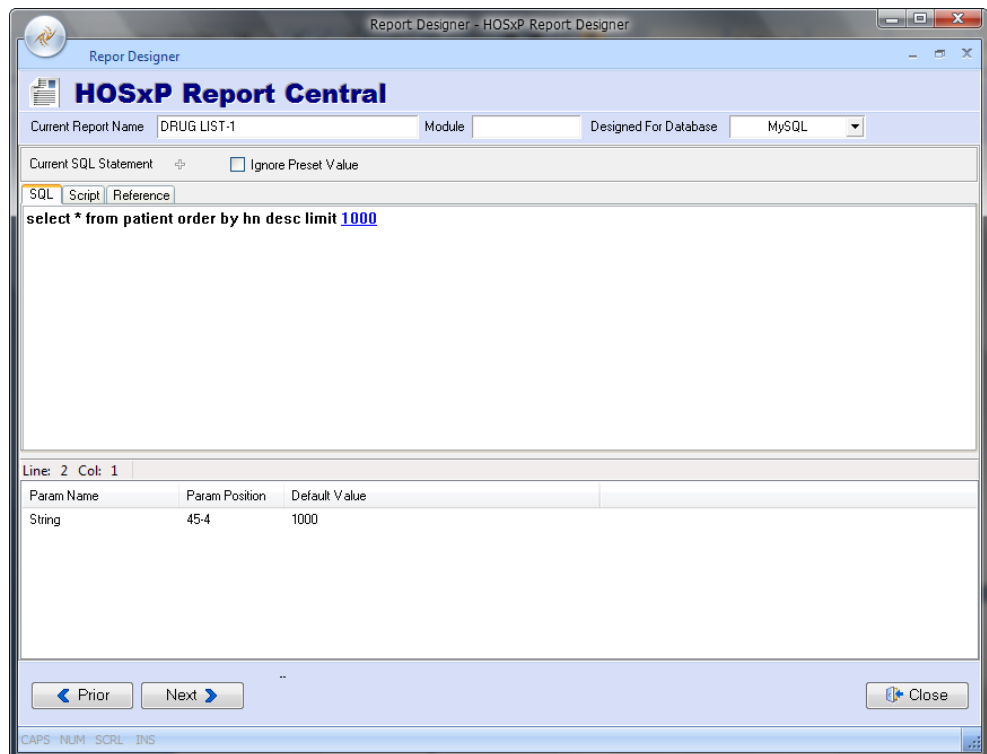
วิธีการสร้างรายงานแบบ sub report สามารถทำได้ดังนี้

เริ่มต้นคำสั่งด้วยการ Select เฉพาะข้อมูลทั่วไปของผู้ป่วยด้วยคำสั่ง

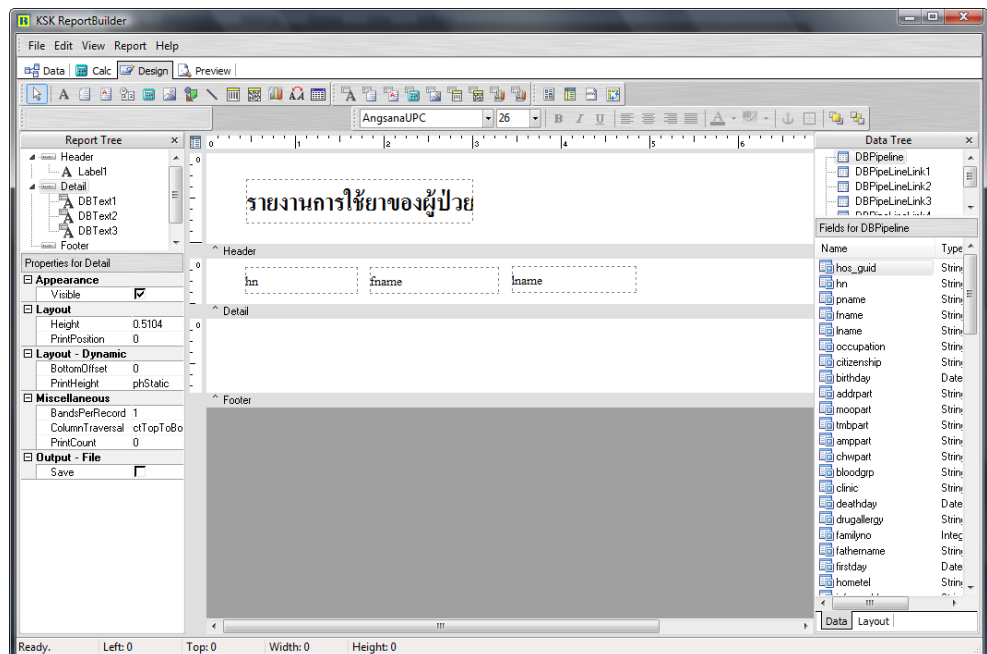
```
select * from patient order by hn desc limit 1000
```

ดังรูป

การใช้ VARIABLE

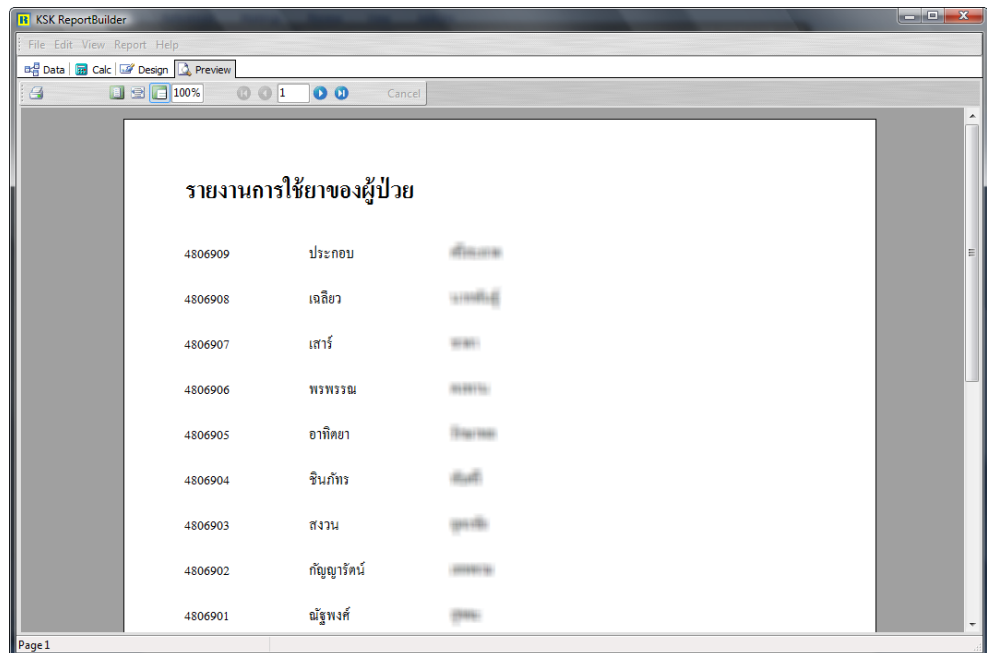


ออกแบบหน้าจอรายงานดังรูป

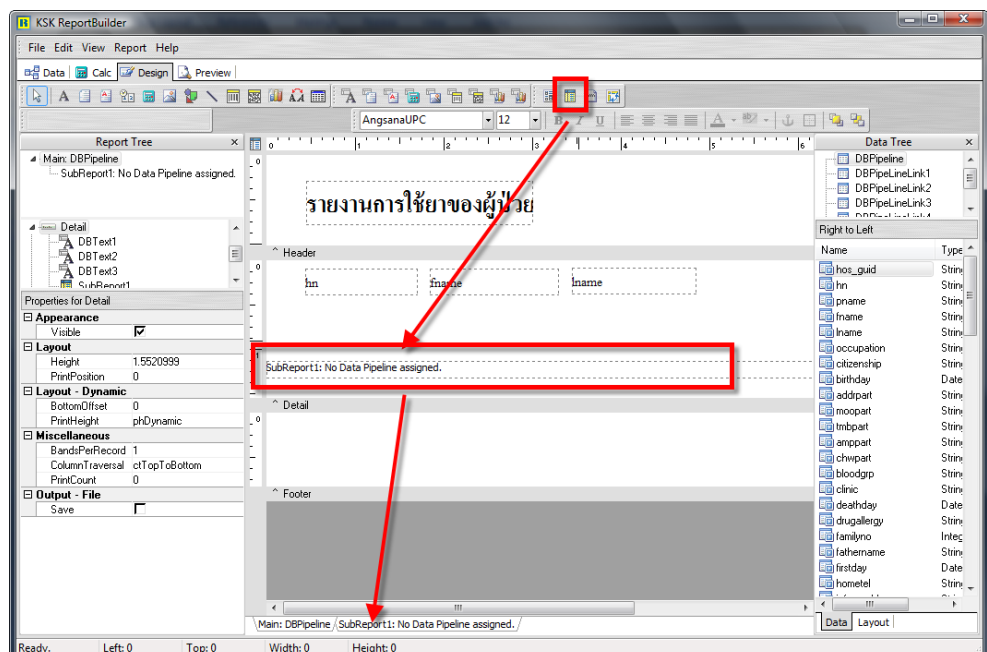


การใช้ VARIABLE

เมื่อลอง Preview ดูตัวอย่างพิมพ์

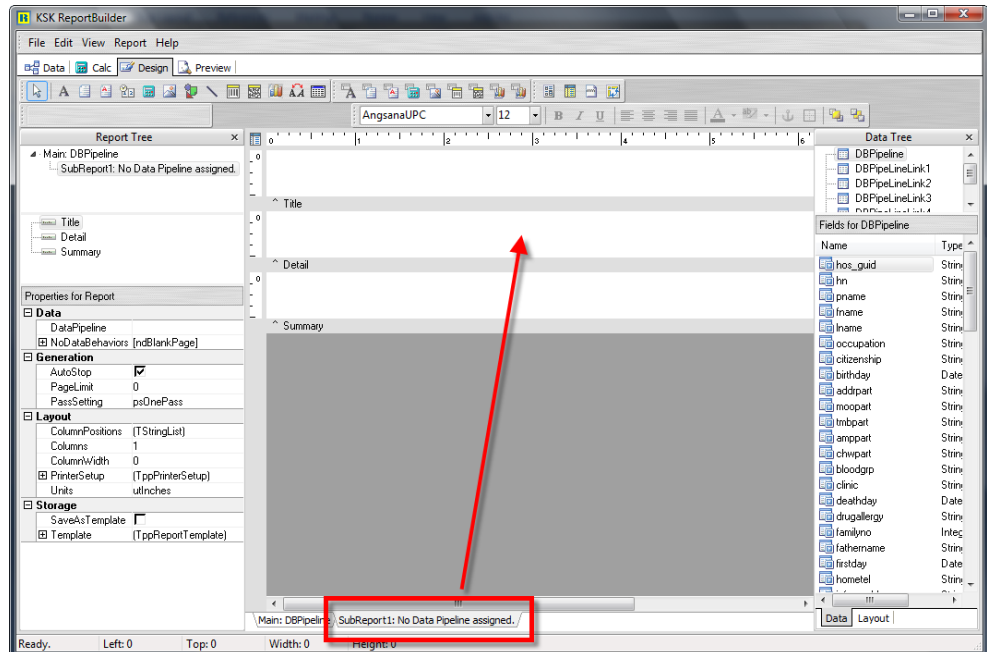


จากนั้นกลับมาที่หน้าออกแบบ ขั้นตอนต่อไปคือการนำข้อมูลรายการยาที่ผู้ป่วยใช้มาใส่ในรายงาน โดยการนำข้อมูลรายการยามาใส่จะนำมาใส่ใน sub report ใน Band detail ดังรูป

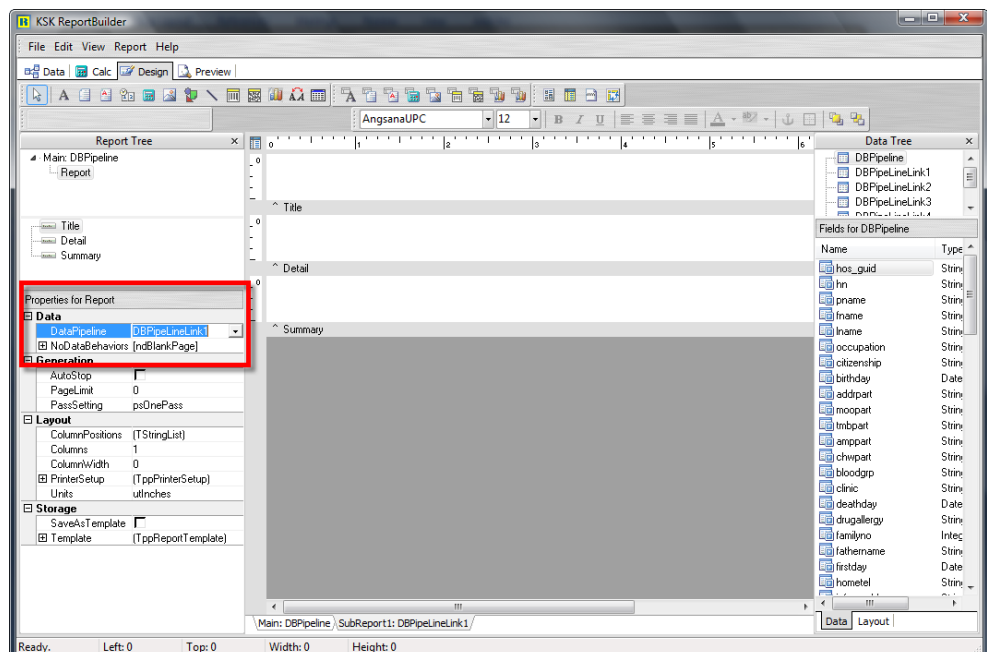


การใช้ VARIABLE

Sub report แต่ละตัวจะมีหน้าต่างออกแบบย่อเป็นของตัวเอง โดยจะแสดงเป็นอีก 1 tab ด้านล่างดังรูป

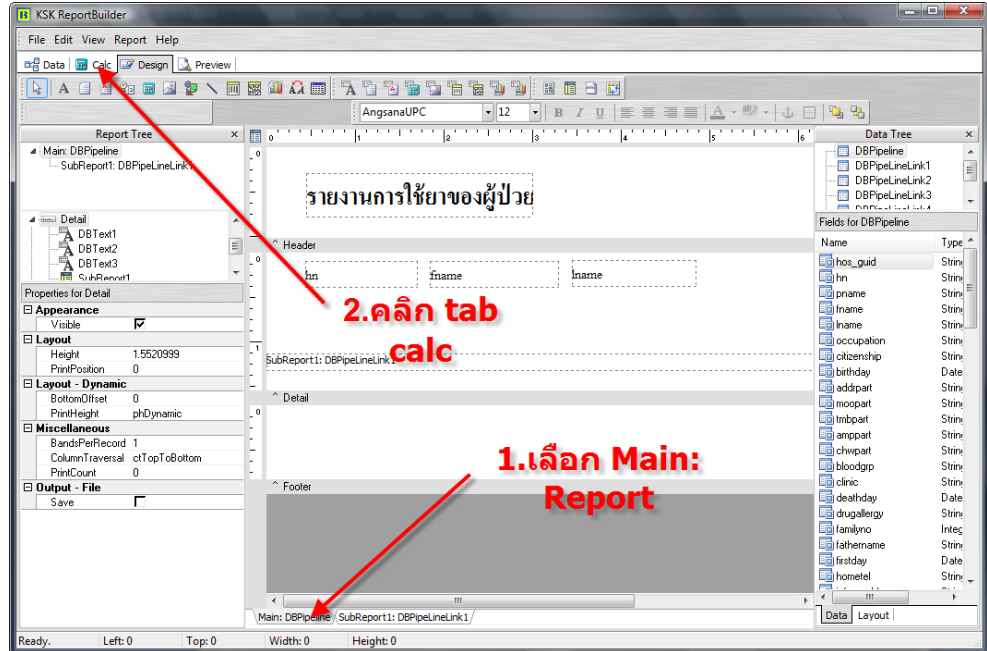


และในแต่ละ Sub report สามารถเลือก associate กับ datapipeline ที่ไม่ซ้ำกันได้ (โปรแกรมมี Datapipeline ให้เลือกใช้ 5 ตัว) การกำหนด Datapipeline ให้กับ Sub report สามารถทำได้ โดยการเลือกจาก Properties ด้านซ้ายมือดังรูป (ในตัวอย่างเลือก DatapipelineLink1)

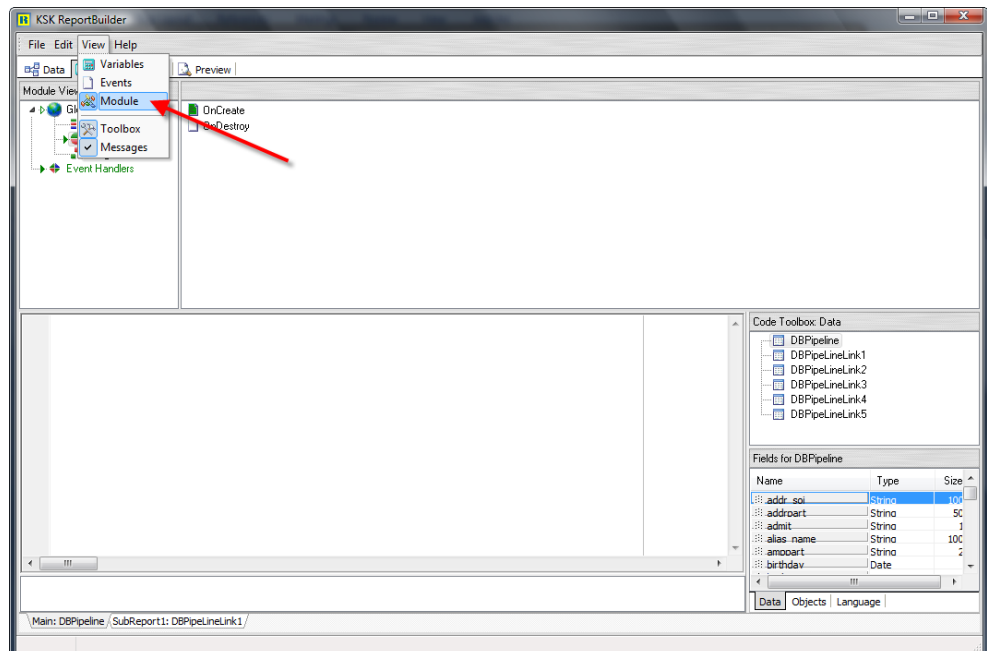


การใช้ VARIABLE

การนำข้อมูลมาใส่ใน DatapipelineLink1 ในขั้นตอนการออกแบบนั้นจำเป็นจะต้องเขียนเป็นชุดคำสั่งใน Report โดยเขียนใน Event OnCreate ของ Report ตามขั้นตอนดังรูป



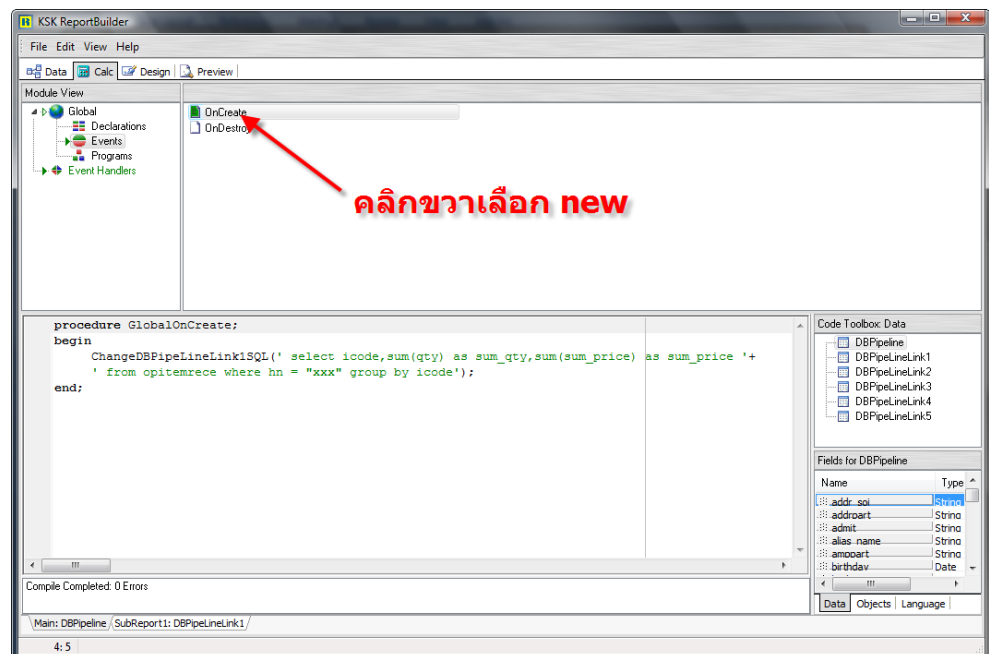
เลือกเมนู View -> Module



การใช้ VARIABLE

คลิกขวาที่ OnCreate แล้วเลือก New จากนั้นเขียน คำสั่งลงในช่องด้านล่างดังนี้

```
procedure GlobalOnCreate;
begin
    ChangedBPPipelineLink1SQL('select icode,sum(qty) as '+
    ' sum_qty,sum(sum_price) as sum_price '+
    ' from opitemrece where hn = "xxx" group by icode');
end;
```

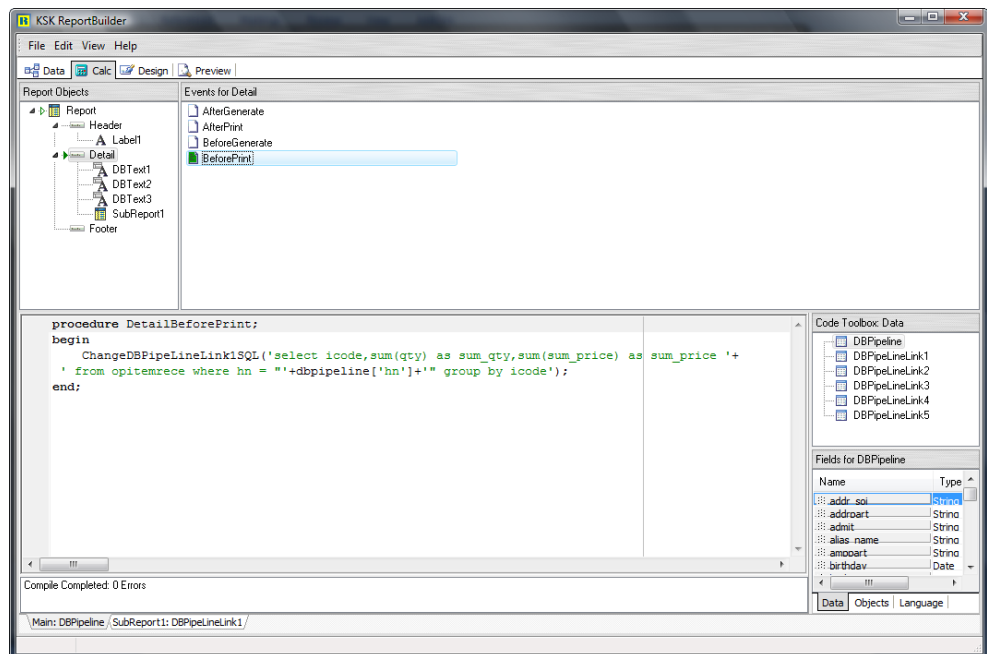


การใช้ VARIABLE

จากนั้นเลือกเมนู View – Event

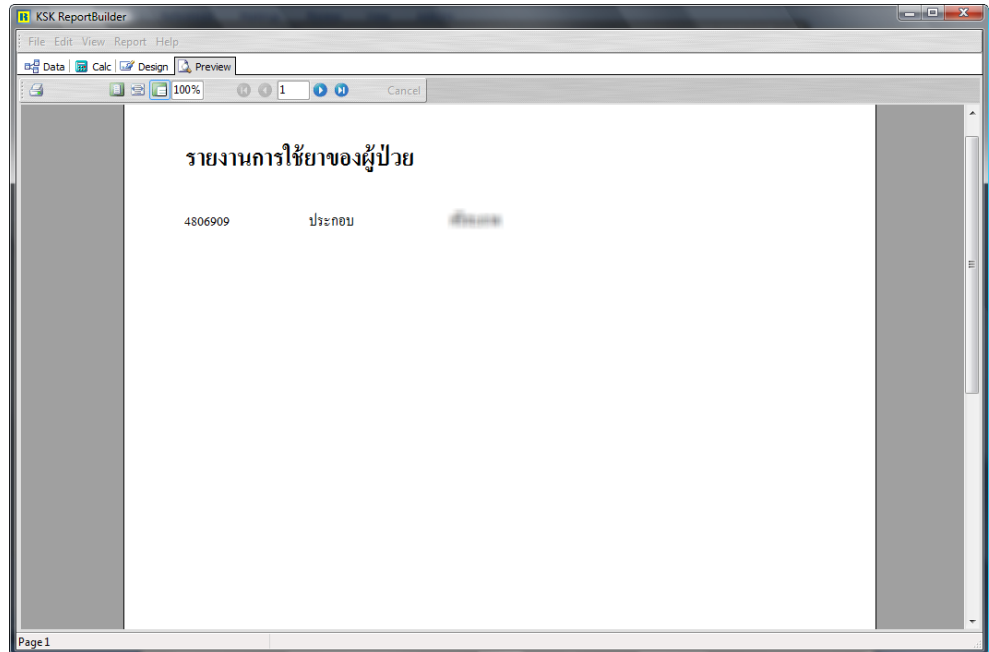
แล้วเลือก Band Detail คลิกขวาตรง BeforePrint แล้วเขียนชุดคำสั่งดังนี้

```
procedure DetailBeforePrint;  
  
begin  
  
    ChangedBPIPELineLink1SQL('select icode,sum(qty) as '+  
  
' sum_qty,sum(sum_price) as sum_price '+  
  
' from opitemrece where hn = '"+dbpipeline['hn']+"' '+  
  
' group by icode');  
  
end;
```

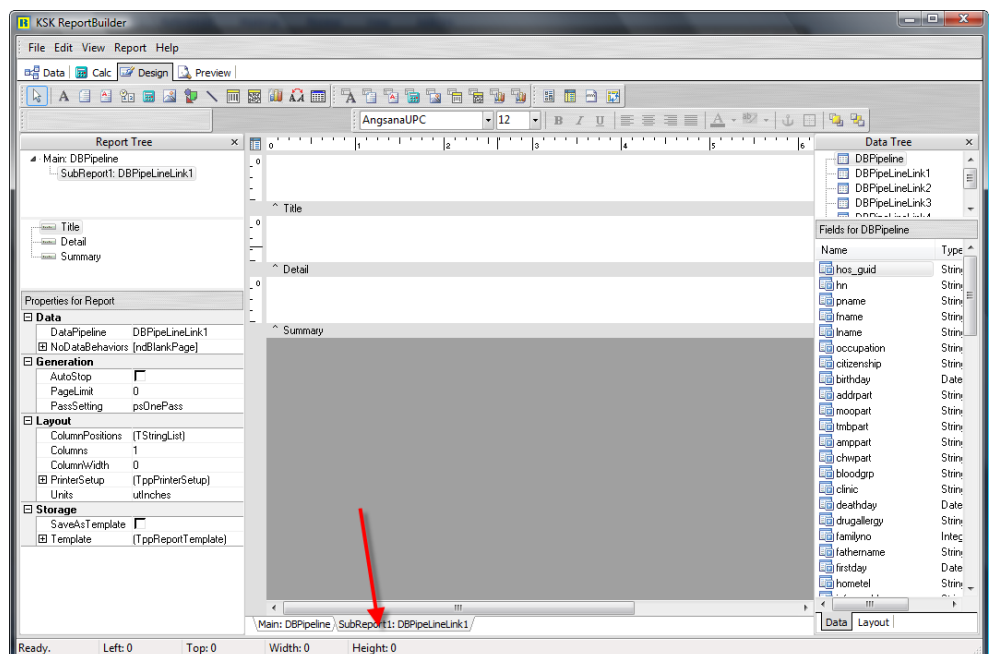


การใช้ VARIABLE

คลิกที่ Tab Preview ก่อน 1 ครั้ง

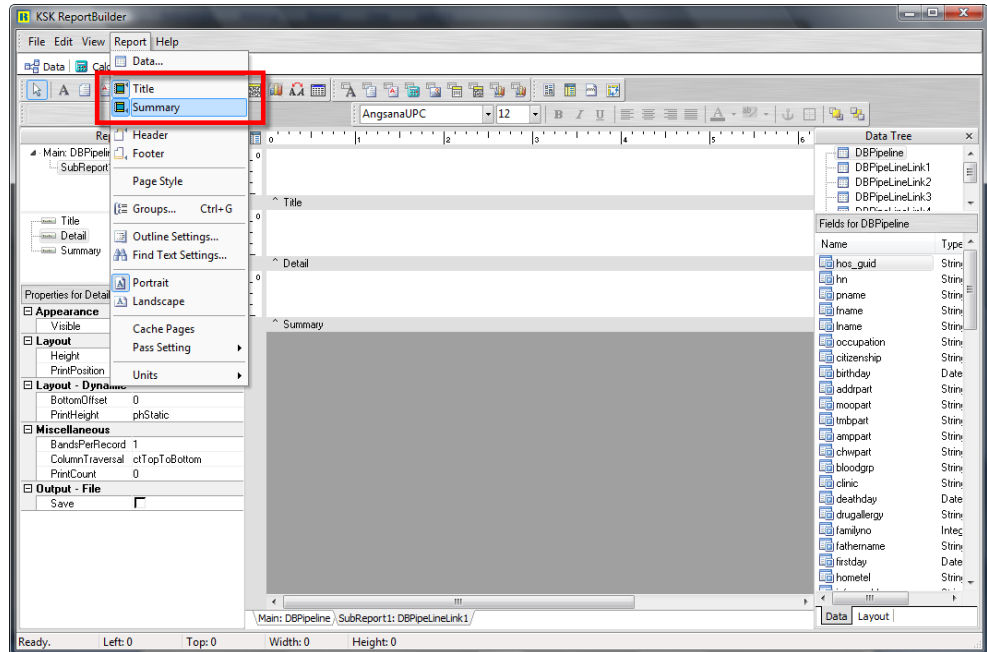


จากนั้นกลับมาที่ Tab design แล้วเลือก Sub report 1

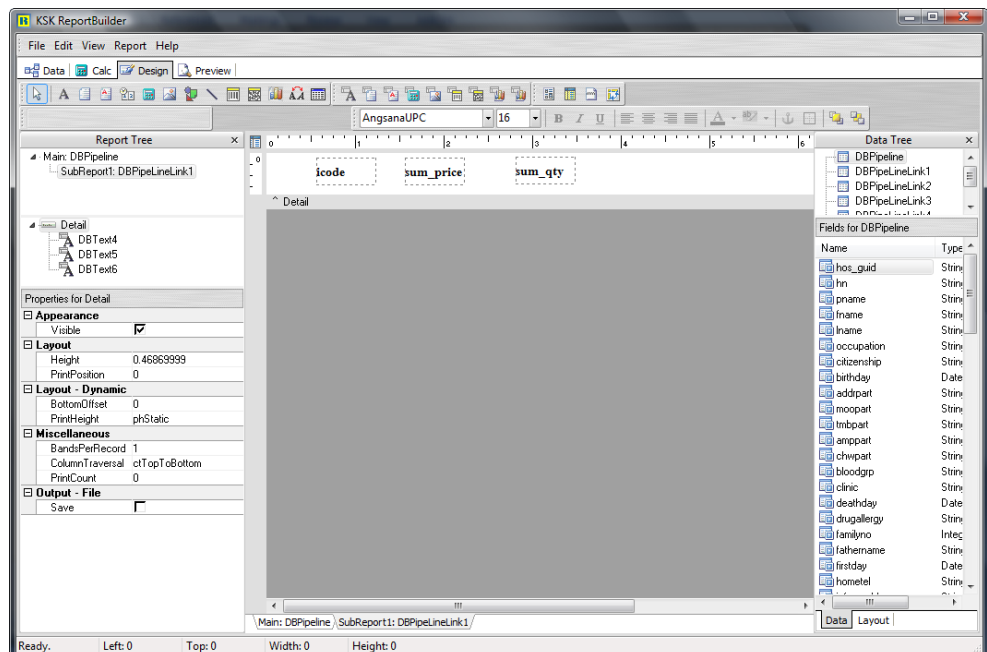


การใส่ VARIABLE

ลบ Band title และ summary ออก

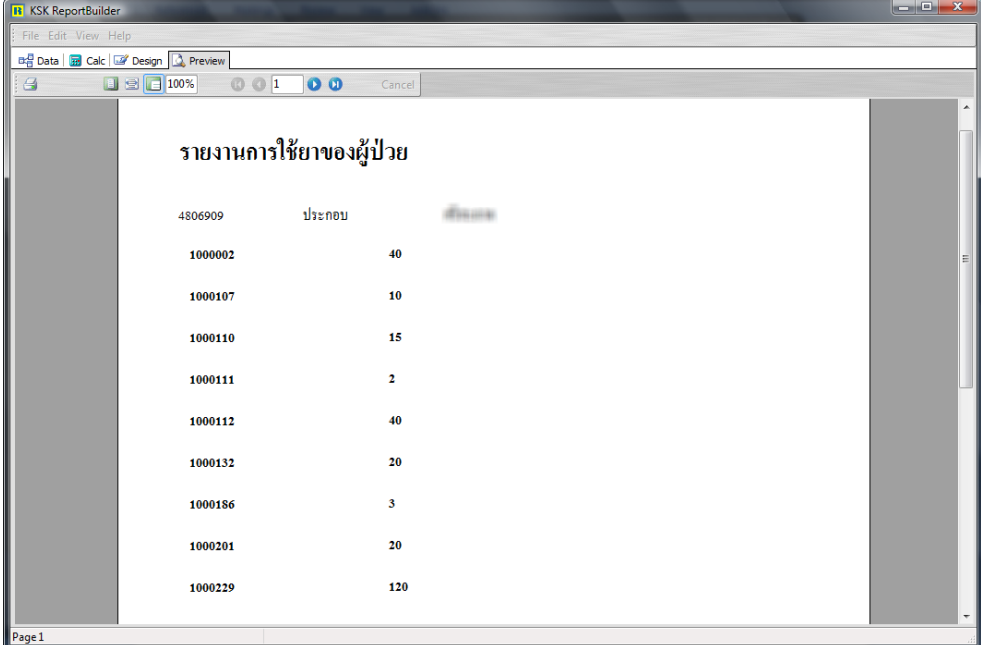


วาง DBText ลงไปแล้วเลือก Field icode sum_price และ sum_qty มาแสดงคั่งรูป



การใช้ VARIABLE

ลอง Preview Report คู่อีกครั้ง จะเห็นว่ามีข้อมูลรายการ (รหัส) ยาที่ผู้ป่วยเคยใช้ทั้งหมด แสดงขึ้นมาด้วย



The screenshot shows the KSK ReportBuilder interface. The main content area displays a report titled "รายงานการใช้ยาของผู้ป่วย" (Patient Medication Report). The report contains a table with the following data:

รหัสยา	ประกอบ	ปริมาณ
4806909	ประกอบ	
1000002		40
1000107		10
1000110		15
1000111		2
1000112		40
1000132		20
1000186		3
1000201		20
1000229		120

The interface includes a menu bar (File, Edit, View, Help), a toolbar with options like Data, Calc, Design, and Preview, and a status bar at the bottom indicating "Page 1".

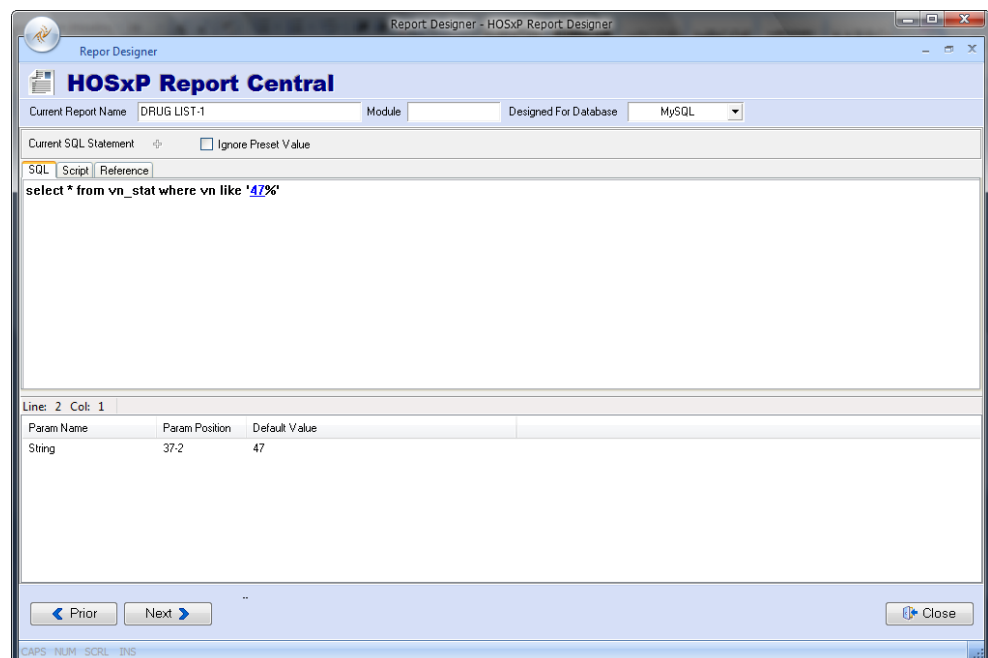
การสร้าง Crosstab report

Crosstab report เป็นรายงานที่นำข้อมูลมาแสดงผลในแบบหลายมิติ ขึ้นอยู่กับผู้ออกแบบว่าต้องการนำเสนอข้อมูลในแต่ละมิติให้สัมพันธ์กันอย่างไร ส่วนใหญ่จะใช้สำหรับทำรายงานสรุปข้อมูลต่างๆ

ตัวอย่างการทำรายงาน Crosstab สรุปค่าใช้จ่ายที่แพทย์ตรวจรักษาผู้ป่วยแยกตามสิทธิการรักษา

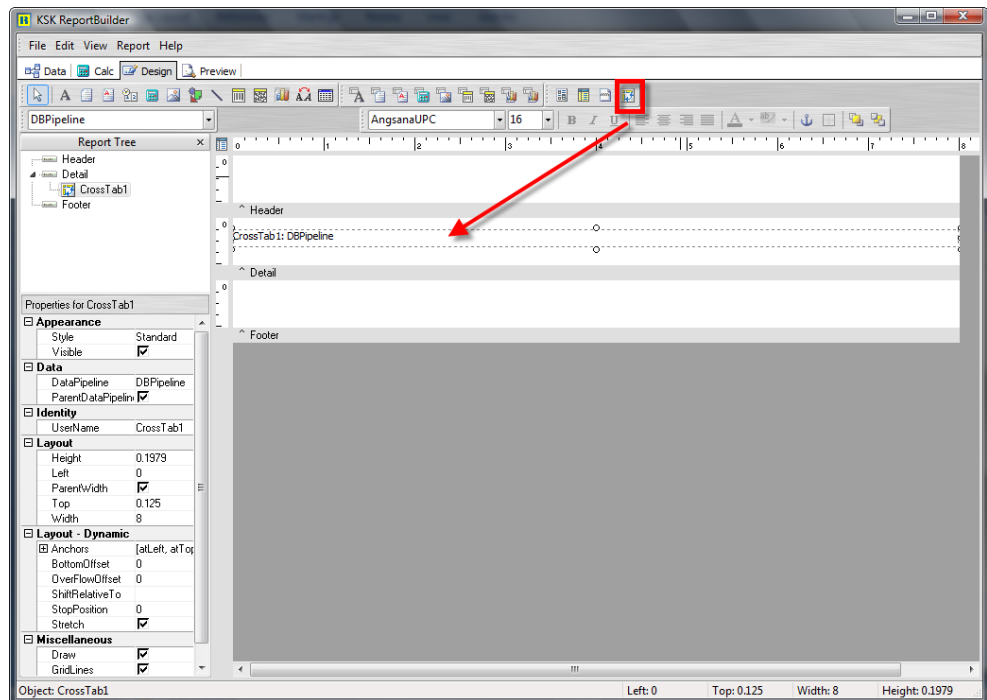
คำสั่งหลักของรายงานเป็นดังนี้ (ข้อมูลทั้งหมดของปี 2547)

```
select * from vn_stat where vn like '47'
```

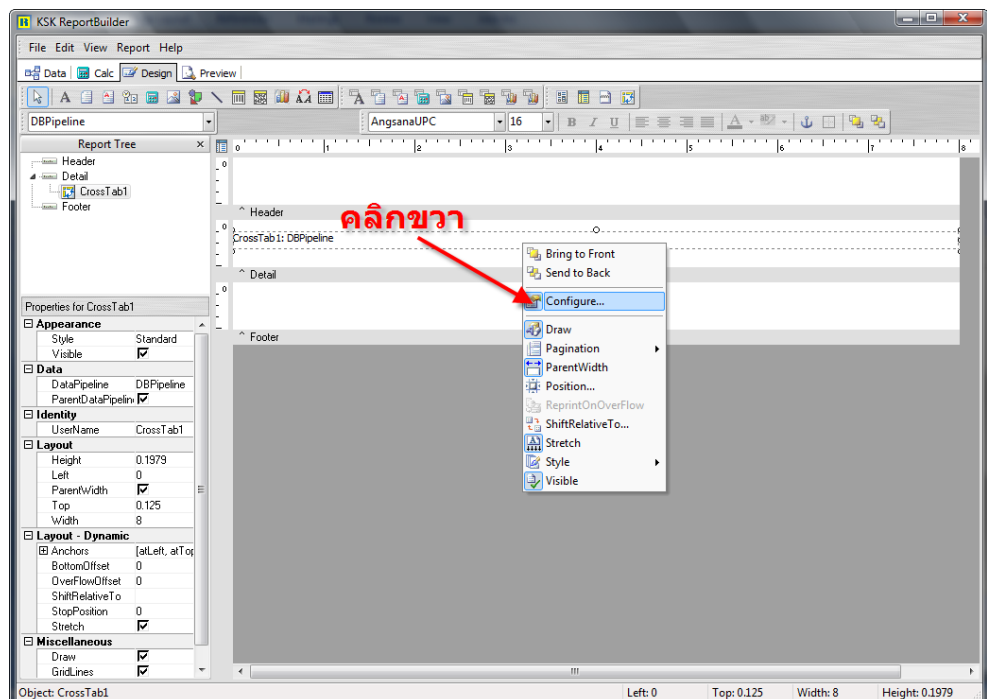


การสร้าง CROSTAB REPORT

ที่หน้าจออกแบบรายงาน คลิกที่ Crosstab component จากนั้นนำมาวางไว้ใน Band detail

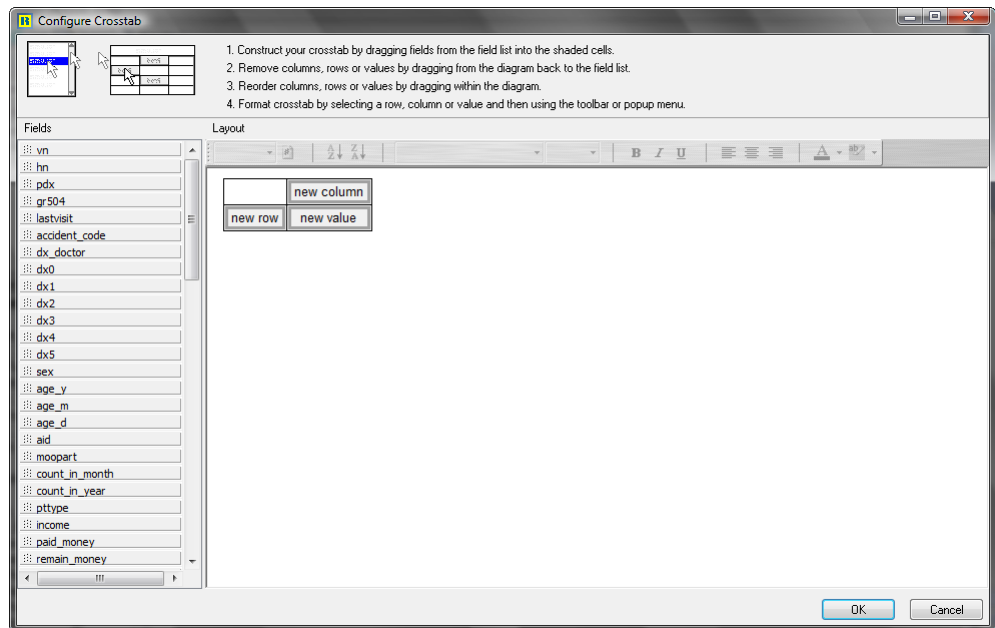


คลิกขวาที่ CrossTab component แล้วเลือก Configure จาก Popup menu

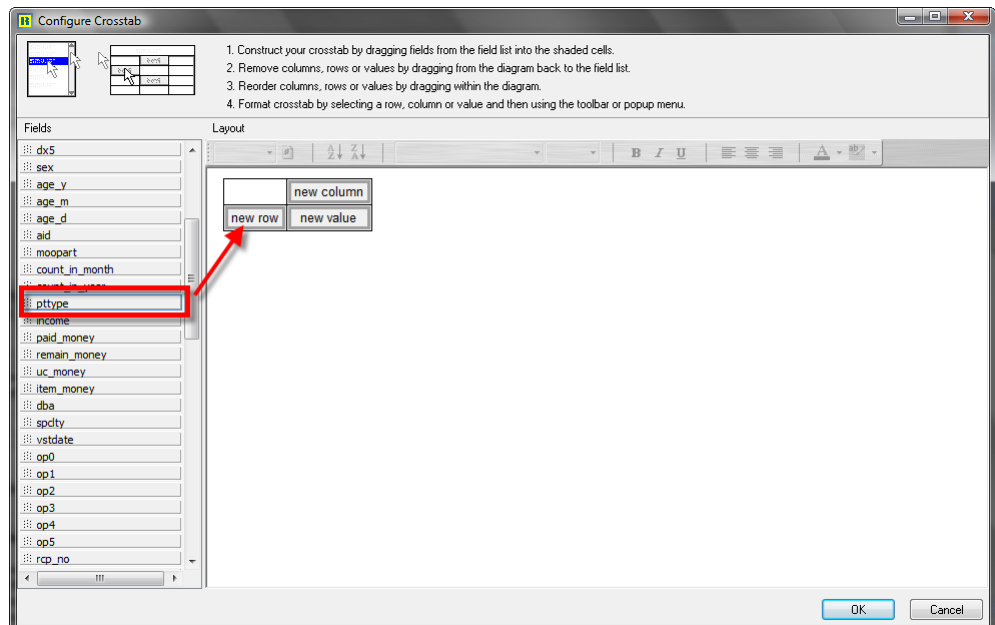


การสร้าง CROSTAB REPORT

หน้าต่างออกแบบ Crosstab report

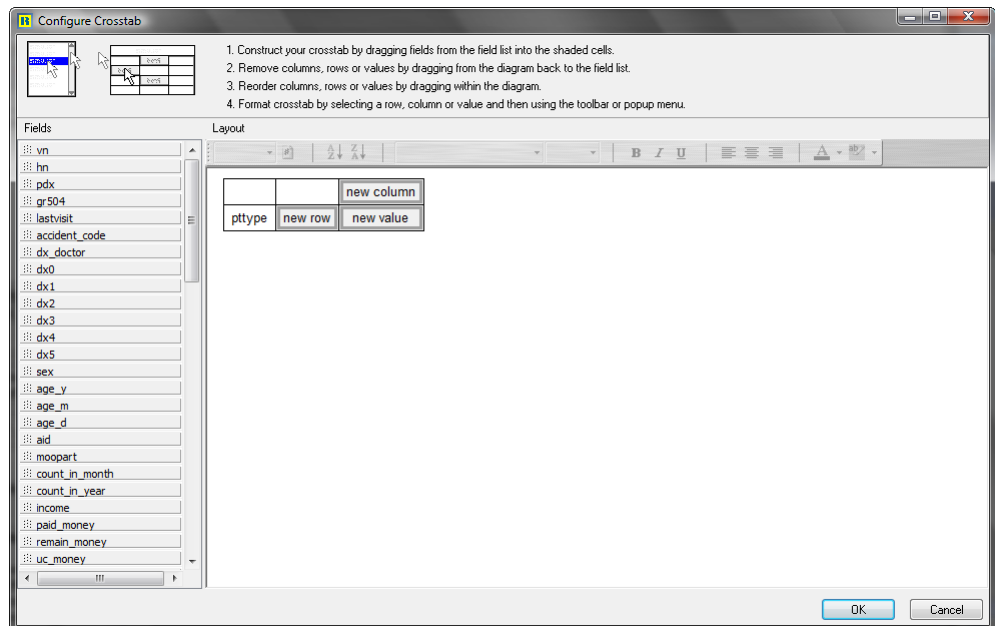


ทำการลาก field ptype จาก field list ทางช่องด้านซ้ายมือมาไว้ในช่อง new row ดังรูป

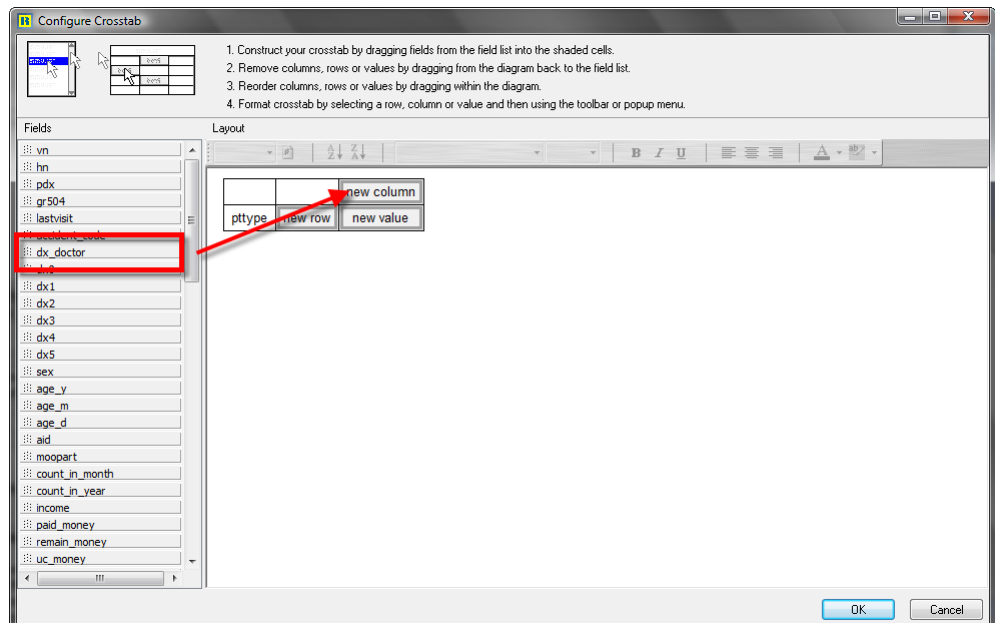


การสร้าง CROSTAB REPORT

หลังจากลาก field ptype มาวางแล้ว

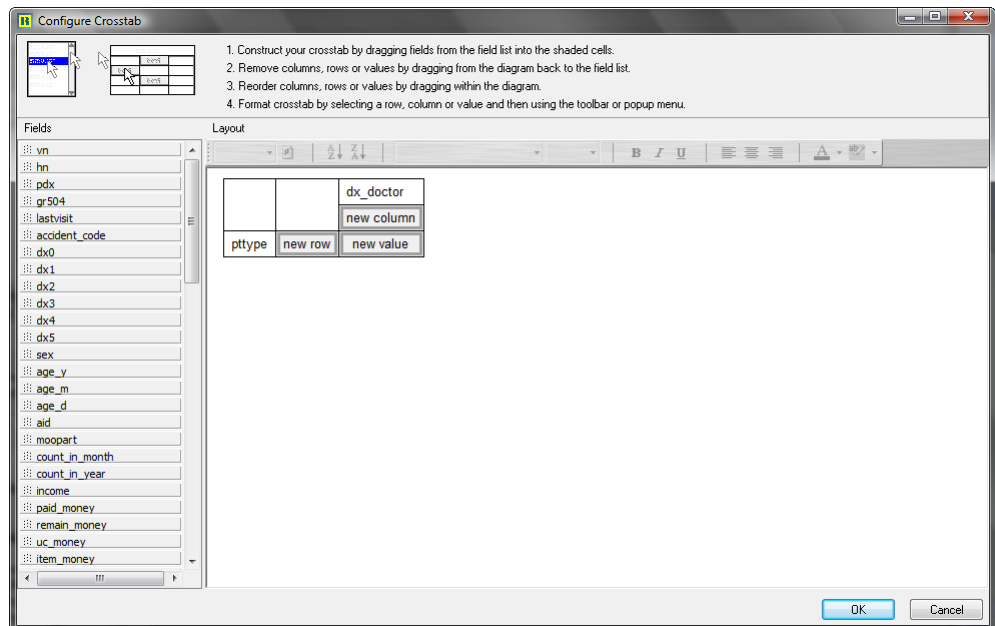


จากนั้นลาก field dx_doctor มาไว้ในช่อง new column ดังรูป

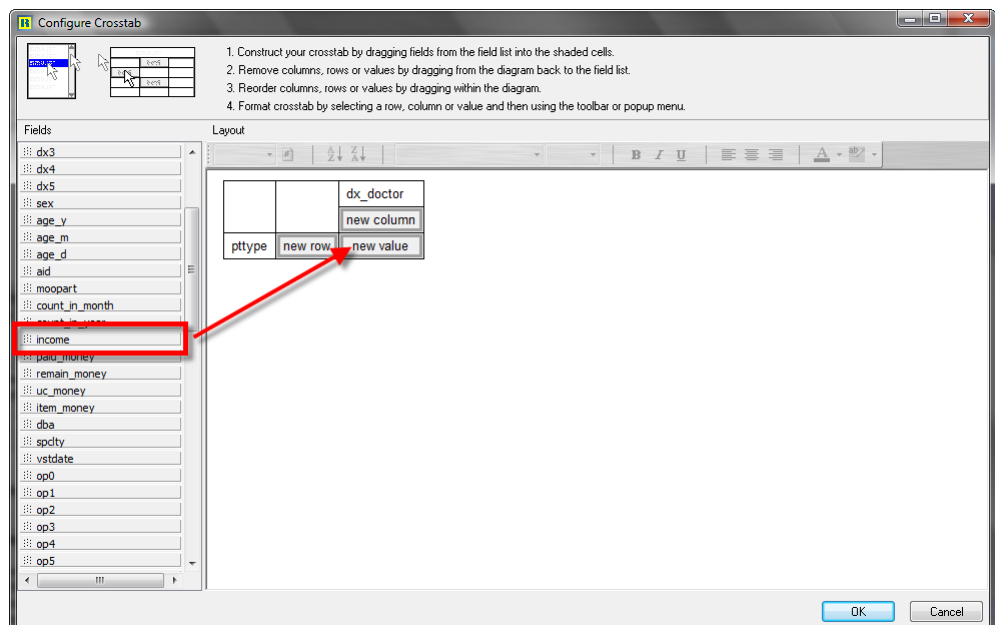


การสร้าง CROSBTAB REPORT

หลังจากลาก dx_doctor มาวางแล้ว

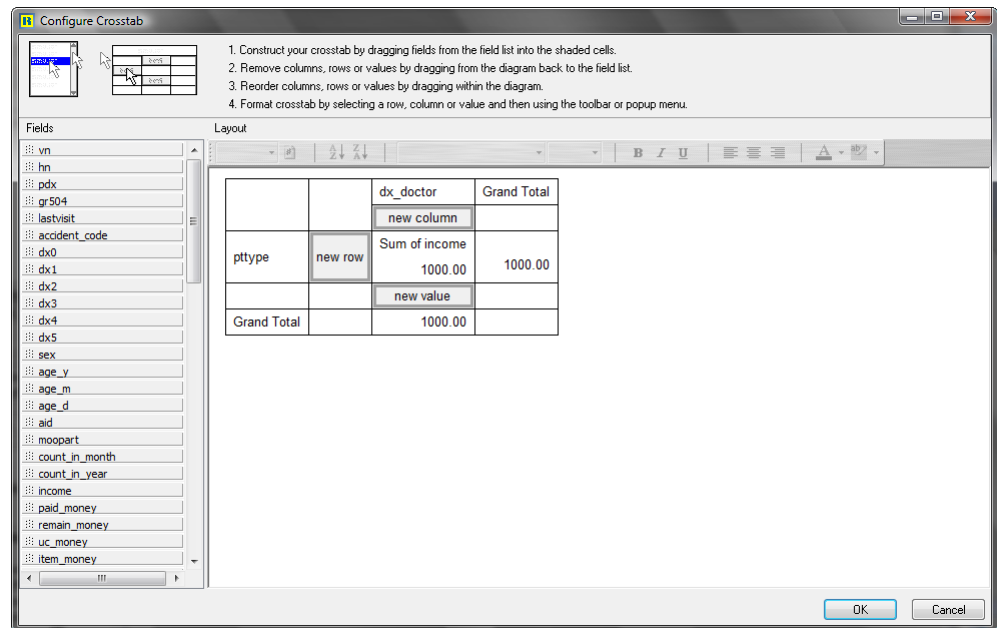


จากนั้นลาก field income มาไว้ใน ช่อง new value ดังรูป



การสร้าง CROSTAB REPORT

ตัวอย่างหลังจากวาง field income แล้ว



จากนั้นกดปุ่ม ok แล้วคลิกที่ tab preview จะได้ผลดังรูป

Sum of income	dx_doctor											
ptype	.91	01	02	03	04	05	06	07	08	09	10	11
10		1110		1255	3042			1715.5	292	951.5	1827.5	2174
20		5641		8038.5	43022	285		5178	251	1347	2856.5	180
30												
33												
34		180		5903	2390.5			14045.5	945	7541	1918	640
36											80	
37												
38		105		3750.25	102	143.5	80	2436	140	430	839	1120
41												
42								900	169		100	
43												
44												
50												
52												
61												
62												
63					110.5				03			

รายงานที่ได้จะแสดงรหัสสิทธิการรักษาไว้ทางด้านซ้าย และ รหัสแพทย์ด้านบน ข้อมูลในแต่ละแถวจะเป็นผลรวมของค่าใช้จ่ายที่แพทย์แต่ละท่านสั่งใช้แยกตามสิทธิการรักษา

การสร้าง Custom Report

Custom Report คือรายงานที่ผู้ใช้งาน หรือ ผู้ดูแลระบบได้ทำการสร้างขึ้นมาใช้งานเอง เนื่องจากรายงานบางอย่าง ทางผู้พัฒนาอาจจะยังไม่ได้สร้างไว้ให้ ซึ่งการสร้าง Custom Report ต้องมีความรู้พื้นฐาน ได้แก่

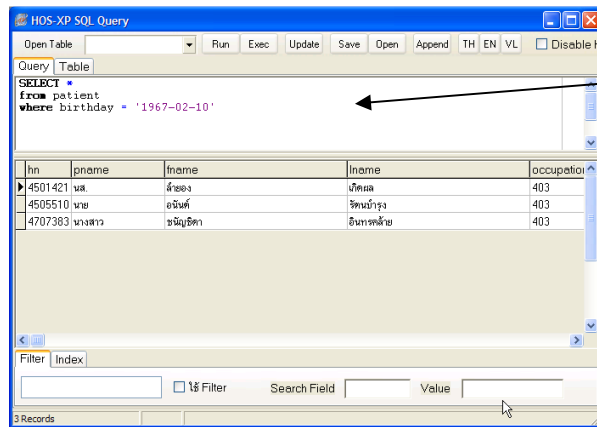
- ความรู้ความเข้าใจในข้อมูลที่ถูกจัดเก็บโดยศึกษาจาก Data Dictionary และ ER Diagram ของระบบ
- ความรู้ความเข้าใจในการใช้งานชุดคำสั่ง SQL เพื่อใช้ในการดึงข้อมูลที่ถูกเก็บไว้ในฐานข้อมูล
- ความรู้ความเข้าใจในการใช้งานระบบ Report Designer

ส่วนขั้นตอนในการสร้าง Custom Report จะมีดังนี้

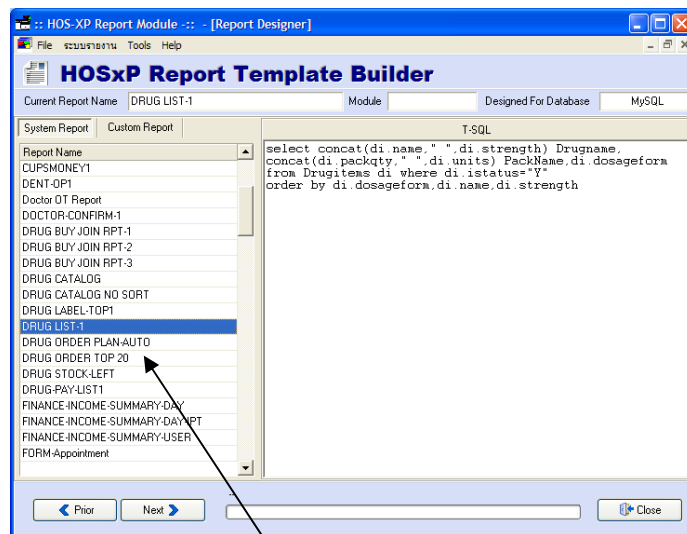
1. สร้างชุดคำสั่งที่ใช้ในการดึงข้อมูล
 2. นำชุดคำสั่งที่ได้สร้างไว้ ไปออกแบบ Report ในระบบ Report Designer
 3. กำหนดช่วงของข้อมูลที่ต้องการให้แก้ไขได้ในขณะทำรายงาน เช่น วันที่, HN, อื่นๆ
 4. ออกแบบรายงาน
 5. ตั้งชื่อรายงานที่ต้องการบันทึก โดยให้ใส่คำว่า CUSTOM- ไว้หน้าชื่อรายงาน
 6. บันทึกรายงาน
-

การ DOWNLOAD รายงาน

ตัวอย่างแสดงการนำข้อมูลที่ได้จากคำสั่ง SQL มาออกแบบเป็น Custom Report

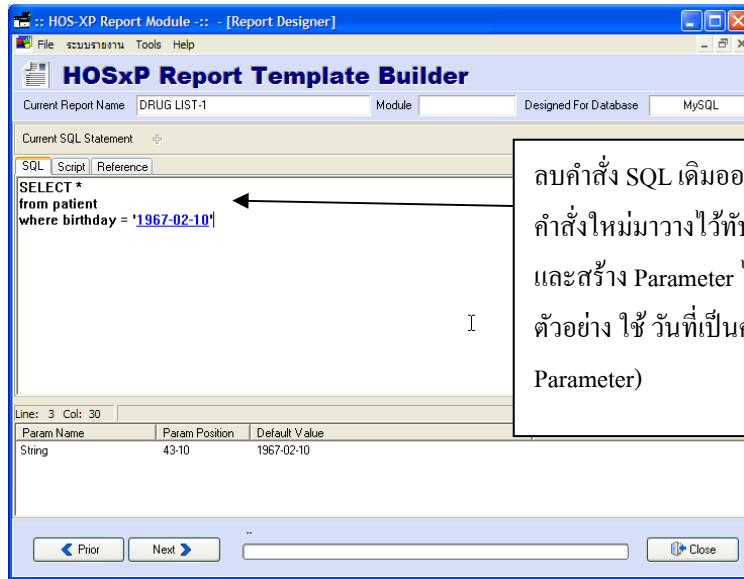


สร้างและทดสอบชุดคำสั่ง
ที่จะทำงานในหน้าจอ
SQL Query
จากนั้น copy คำสั่งที่ใช้ไว้
ใน clipboard (Ctrl+C)

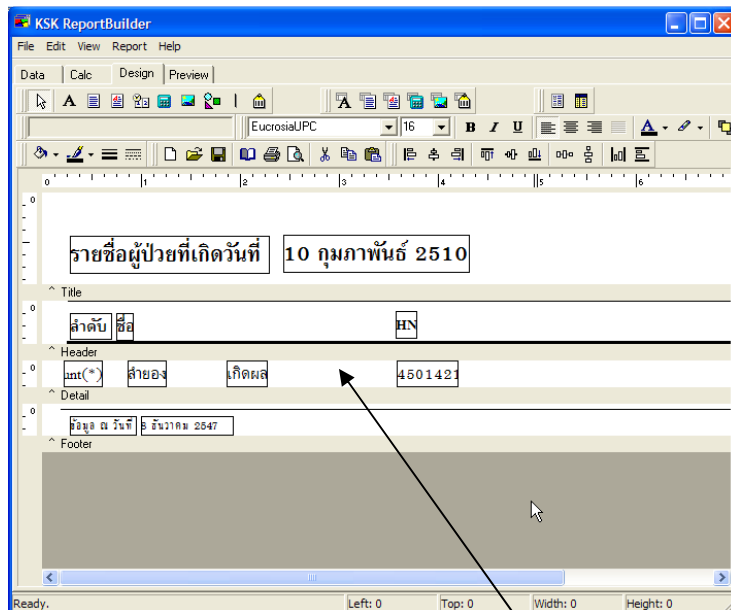


เข้าระบบรายงาน และเลือกรายงานใดๆ ก็ได้ที่
ต้องการใช้เป็นแม่แบบ

การ DOWNLOAD รายงาน

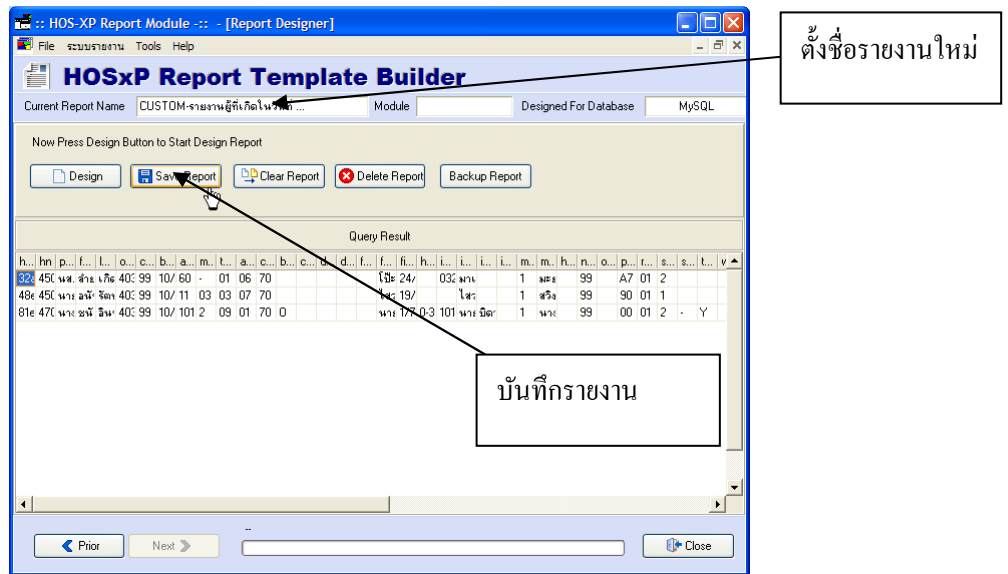


ลบคำสั่ง SQL เดิมออก และนำคำสั่งใหม่มาวางไว้ที่บับของเดิม และสร้าง Parameter ไว้ (จากตัวอย่าง ใช้ วันที่เป็นค่า Parameter)

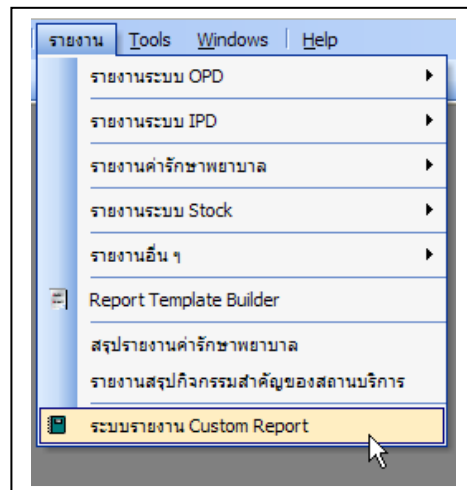


ออกแบบรายงาน ให้เรียบร้อย

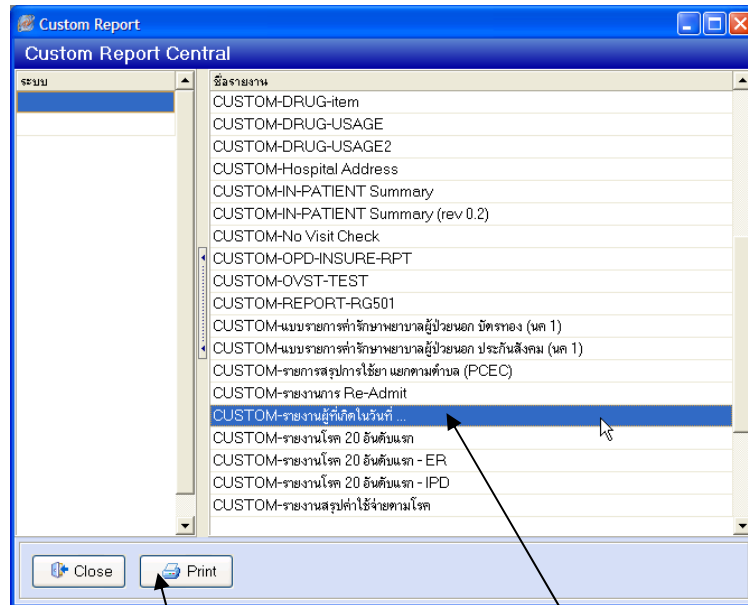
การ DOWNLOAD รายงาน



เมื่อได้ทำการสร้าง Custom Report ไว้แล้ว สามารถเรียกมาใช้งานได้จาก โปรแกรม HOSxP โดยเลือกจากคำสั่ง Custom Report ดังรูปด้านล่าง

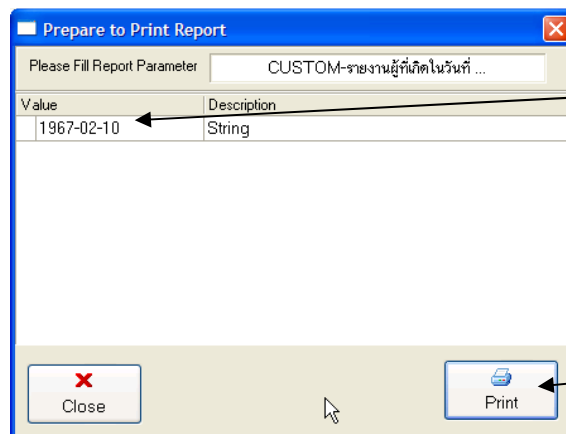


ms DOWNLOAD รายงาน



กดปุ่ม Print เพื่อเลือก
รายงานมาแสดง

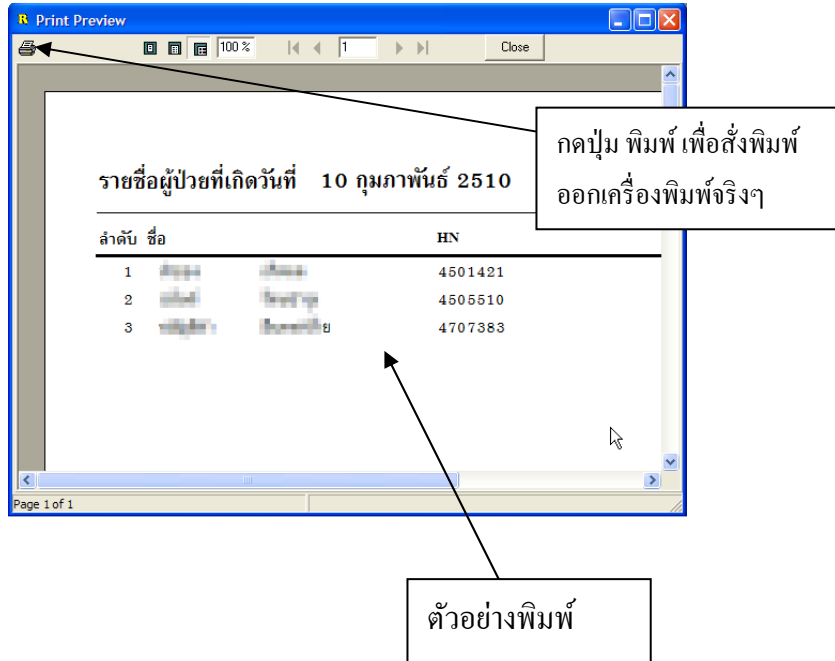
รายงานที่ได้สร้างไว้จะปรากฏให้เลือก



ข้อมูลที่ตั้งไว้ใน
Parameter สามารถแก้ไข
ได้

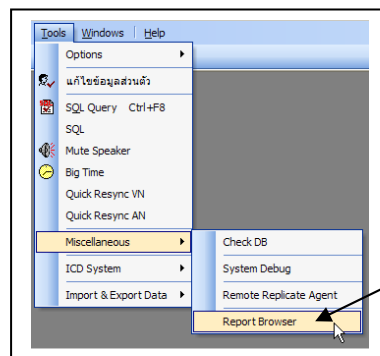
กด Print เพื่อทำการ
แสดงตัวอย่างรายงาน

การ DOWNLOAD รายงาน

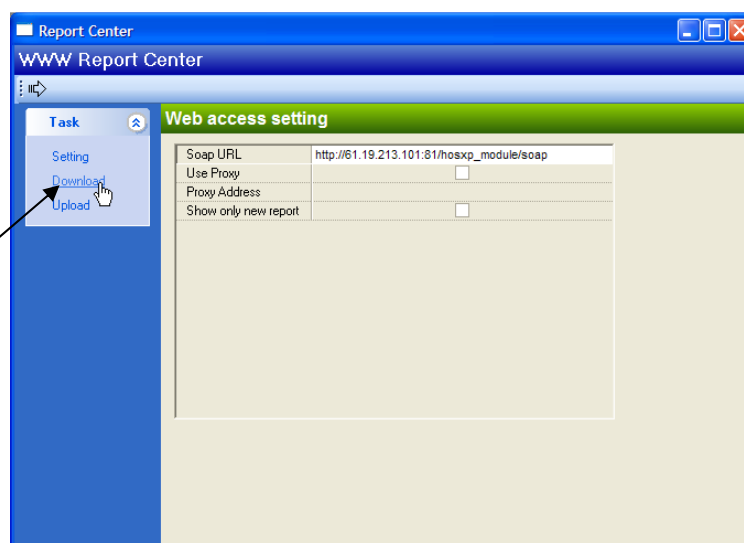


การ Download รายงานใหม่ๆ จาก Web Report Central

HOSxP มีรายงานใหม่ๆ ออกมาอยู่เสมอ ผู้ดูแลระบบสามารถทำการ Download รายงานใหม่ๆ มาติดตั้งและใช้งานได้เลย ผ่านระบบ Report Browser โดยสามารถทำได้ตามขั้นตอนดังนี้

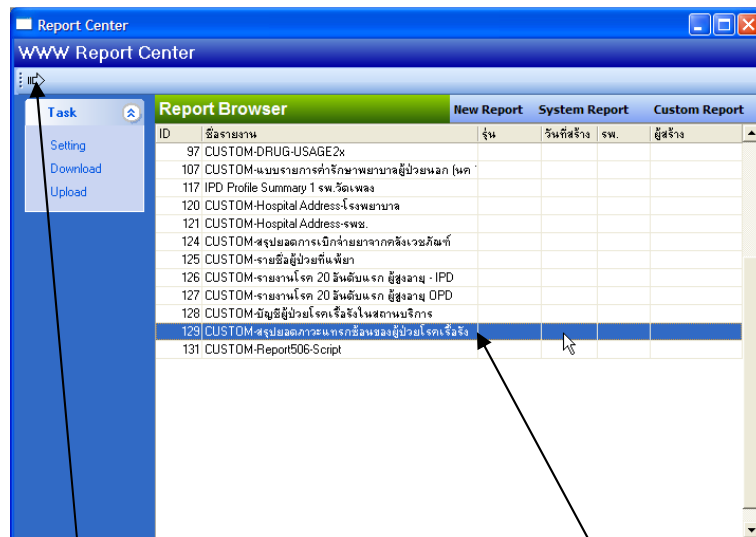
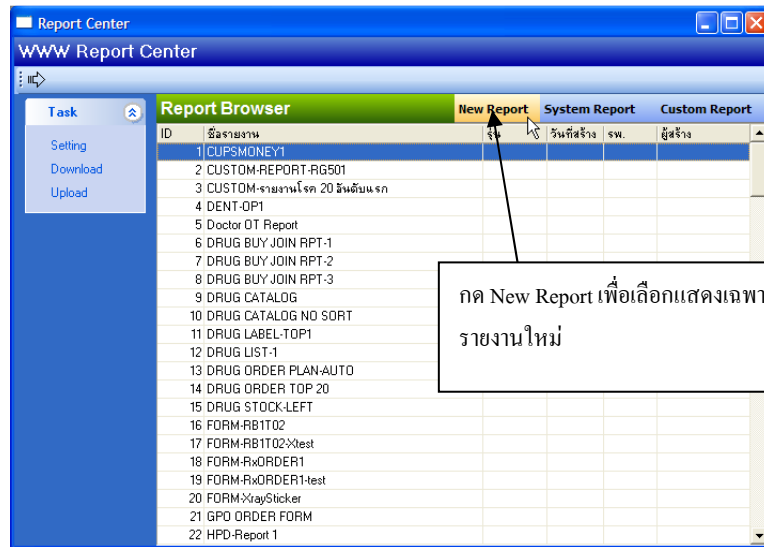


เลือกคำสั่ง Report Browser จาก
เมนู Tools -> Miscellaneous

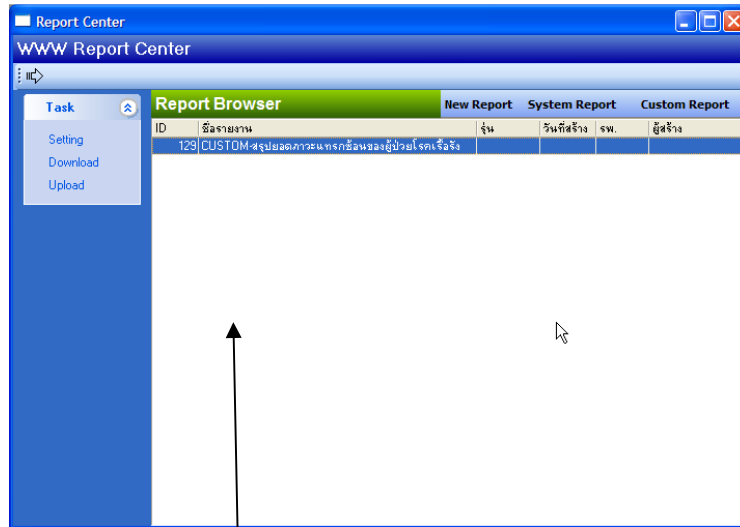


เลือกคำสั่ง
download จาก
Sidebar

ms DOWNLOAD รายงาน



การ DOWNLOAD รายงาน



Download ได้สำเร็จ

