

## New Generic Queries

The PS Query Users Group, a committee sponsored by the New York State Payroll Users Group, has designed new generic queries and added them to the public query section of PS Query. Generic public queries make it easier for PS Query users to retrieve information needed for frequently used reports or data requests.

The new queries are grouped into five categories. Each query is described below, followed by screen prints of the fields and criteria included in each query definition.

The names of generic queries start with “00-” (zero zero hyphen). They appear at the top of the public query list.

<u>Category</u>	<u>Screen Prints on Pages</u>
General Interest Queries	5 – 11
Archived Data Queries	12 – 15
“Difference” Queries	16 – 17
Duplicate Overtime Payment Queries	18 – 20
Position Management Queries	21 – 24

<b>Category 1 - General Interest Queries</b>		
The queries in this section cover a wide range of topics – staffing, additional pay, deductions, miscellaneous payments and mail drop ID (check sorting).		
Query Name	Purpose	Notes
<a href="#">00-ActiveStaff</a>	List of currently active staff	Prompt on DeptID
<a href="#">00-AddlPay</a>	List of staff currently receiving additional payments.	The following earnings codes are included: LOC, SDF, GEO, SED, PS1. The query can be modified to include other codes as needed.*
<a href="#">00-DeductionCodePrompt</a>	Returns a list of employees who have a specific deduction.	Prompt for a valid Deduction Code.
<a href="#">00-Earnings By FundCode</a>	Identifies the Funding Code associated with each Earnings Codes in a specific pay period.	This query takes 10 – 15 minutes to return results.
<a href="#">00-IndividualHistory</a>	Returns all rows entered in the Job table for an individual employee while in your agency.	Prompt on the EmplID field.
<a href="#">00-OT ByPayPeriod</a>	Returns rows for all Earnings Codes that start with the letter “O” in a specific pay period.	The criteria can be modified to list specific Earnings Codes. There are some Earnings Codes that start with the letter “O” that are not overtime payments, e.g. OVP for Overpayments.*
<a href="#">00-Pay ByMailDrop</a>	Returns a list of employees paid in a specific pay period. The list is sorted by the Mail Drop (check sort) field.	Employees who receive both a check and an advice are listed twice. Checks are identified by the name of the bank issuing the check, currently “Fleet”.

<b>Category 2 - Archived Data Queries</b>		
<p>The queries in this section return archived earnings information. Earnings data is periodically archived or moved from the main earnings tables. Archived earnings data is saved in these tables:</p> <ol style="list-style-type: none"> <li>1. NY_Pay_Check</li> <li>2. NY_Pay_Earnings</li> <li>3. NY_Pay_Ern_Dist</li> <li>4. NY_Pay_Oth_Earns</li> </ol>		
Query Name	Purpose	Notes
<a href="#">00-Arc DeptPayCheckData</a>	Returns data from NY_Pay_Check and NY_Pay_Ern_Dist for a single archived pay period.	Prompt for a valid pay period end date.
<a href="#">00-Arc Dept OtherEarns</a>	Returns archived miscellaneous payment data (e.g. OTA) from the NY_Pay_Earnings and NY_Pay_Oth_Earn tables.	Prompt for a valid pay period end date. This query can be modified to include criteria for specific earnings codes.*
<a href="#">00-Arc Ind OtherEarns</a>	Returns a history of archived miscellaneous payments for a specific employee.	Prompt on the EmplID field.
<a href="#">00-Arc Ind PayCheckData</a>	Returns a history of archived earnings information for a specific employee.	Prompt on the EmplID field.

<b>Category 3 - "Difference" Queries</b>		
The queries in this section return rows where there is different information in two tables. This is not necessarily incorrect but should be reviewed to determine if changes are needed. Some employees prefer different addresses or names in these two tables.		
Query Name	Purpose	Notes
<a href="#">00-DifferentAddresses</a>	Returns rows where the employee's address in the Personal Data table is different from his/her address in the Pay_Check table. This is not necessarily incorrect. Some employees prefer different addresses in these two tables.	Prompt on DeptID
<a href="#">00-DifferentNames</a>	Returns rows where the employee's name in the Pay_Check table is different from his/her name in the Personal Data table. This is not necessarily incorrect. Some employees prefer different names in these two tables.	Prompt for a valid pay period end date.
<b>Category 4 - Duplicate Overtime Payment Queries</b>		
The queries in this section identify duplicate overtime payments. The purpose is to determine if multiple overtime rows have been entered for the same employee for the same day(s).		
Query Name	Purpose	Notes
<a href="#">00-Dup_OT_Audit_Daily</a>	Identifies when multiple overtime rows have been entered for the same employee for the same day(s). Contains a prompt on the data entry date.	Run the 00-Dup_OT_Verification query to verify the results.
<a href="#">00-Dup_OT_Audit</a>	Similar to previous query but contains a prompt on the pay period end date.	Run the 00-Dup_OT_Verification query to verify the results.
<a href="#">00-Dup_OT_Verification</a>	The purpose of this query is to verify if a specific employee listed in either of the two prior queries has received a previous payment for the same days of overtime that are being processed in the current pay period.	Prompt on the EmplID field. It is run for one employee at a time.

<b>Category 5 – Position Management Queries</b>		
Queries in this section relate to Position Management. They return information on funding sources and identify if active positions are either filled or vacant.		
<b>Query Name</b>	<b>Purpose</b>	<b>Notes</b>
<a href="#">00-PositionPool_Funding</a>	Returns a list of agency Position Pools and associated funding information.	Prompt on DeptID
<a href="#">00-Positions_FilledActive</a>	Returns a list of currently filled positions in the agency. It includes data from the Job table associated with the employee currently filling the position.	Prompt on DeptID
<a href="#">00-Positions_VacantActive</a>	Returns a list of currently vacant active positions in the agency.	Prompt on DeptID
<a href="#">00-PS Certificate</a>	This is a PS Query version of the Reveal NHRP723 (PS Certificate) report. The report sorts item numbers by fund, sub-fund and cost center. If run to Excel, the data can then be re-sorted and manipulated as the agency chooses.	The NHRP723 report replaced the PS Schedules report. The new report is sorted differently and it can be more difficult to find an item number without knowing all the associated funding info. This query allows the user to re-sort the data by item number. It can also provide more current data than the monthly NHRP723 report.

### **\*Modifying Public Queries**

Before modifying any Public query, change it to a Private query and save it with a different name. This is particularly important when modifying the generic Public queries that begin with 00-. Modifications to the generic queries could impact a wide range of query users.

1. Open the Public query.
2. Click File on the menu bar and select Properties from the drop down list. (Do this BEFORE making any modifications to the Public query.)
3. Un-check the Public box.
4. Click OK.
5. Click File and select Save As.
6. Type a new name for the query. Public and Private versions of a query MUST be named differently.
7. Click OK. Verify that the query is saved as a Private version. Look at the blue title bar and the query name should be preceded by "PRIVATE.QUERY". If not, return to step 2.
8. Make modifications, as needed, to the Private version of the query.
9. Click File – Save to save your changes.
10. If you do not want to save the changes, click File – New. A message box will ask if you want to save the changes. Click No.

These steps will insure that any PS Query user can run the generic queries and obtain the intended results.

**00-Active Staff**

List of currently active staff and selected fields from Job, Personal Data and JobCode tables.

The screenshot shows a database query window titled "PUBLIC.QUERY.00-ActiveStaff - Query". The window has a menu bar (File, Edit, View, Go, Favorites, Criteria, Help) and a toolbar. On the left is a tree view showing the query structure with tables A, B, and C. The main area displays a table with the following columns: Col, Record.Field, Format, Rel, Ord, Xlt, Agg, and Heading. The data rows are as follows:

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	B.NAME - Name	Char50		1			Name
2	A.EMPLID - EmplID	Char11	X				ID
3	C.DESCR - Description	Char30					Title
4	A.NY_LINE_ITEM_NBR - Line Nu	Char5					Line Nbr
5	A.POSITION_NBR - Position Num	Char8					Position #
6	A.NY_NYS_POSITION - NYS Po	Char8					NYS Position #
7	A.REG_TEMP - Regular/Tempora	Char1			N		Reg/Temp
8	A.FULL_PART_TIME - Full/Part Ti	Char1			N		Full/Part
9	A.GRADE - Salary Grade	Char3	X				Grade
10	A.NY_BARG_UNIT - Bargaining	Char2					Bargain Unit
11	A.NY_EE_FTE_PCT - Employee	Num6.4					Work Pct
12	A.NY_EE_REG_TEMP - Appoint	Char1			S		Appt CD
13	A.EMPL_TYPE - Employee Type	Char1			S		Type
14	A.NY_FTA_SALARY - Full-Time	Num19.6					Annual Salary
15	A.HOURLY_RT - Hourly Rate	Num19.6					Hrly. Rate
16	A.NY_INCREMENT_CD - Increme	Num4.0					Increment
17	A.NY_ANNIVERSARY_DT - NY	Date					Anniversary Date
18	A.DEPTID - Department	Char10	X				DeptID
19	A.NY_PAY_BASIS_CD - Pay Ba	Char3					Pay Basis

The screenshot shows the criteria tab of the same database query window. It displays a table with the following columns: Logical, Expression, Operator, and Expression. The criteria are as follows:

Logical	Expression	Operator	Expression
	A.EFFDT - Effective Date	Eff Date <=	Current Dt (EffSeq = Last)
AND	A.EMPL_STATUS - Employee St	equal to	A
AND	A.DEPTID - Department	equal to	:1
AND	C.EFFDT - Effective Date	Eff Date <=	Current Dt

**00-AddlPay**

List of staff currently receiving additional payments. The following earnings codes are included: LOC, SDF, GEO, SED, PS1. The query can be modified to include other codes used in your agency as needed. Refer to page 3 for instructions on modifying Public queries.

The screenshot shows a window titled "PUBLIC.QUERY.00-AddlPay - Query". The left pane shows a tree view with tables: A = PERSONAL\_DATA - EE P, B = ADDL\_PAY\_DATA - Additi, C = JOB - EE Job History joinec, and D = EARNINGS\_TBL - Earning. The right pane shows the "Fields" tab with a table of field definitions:

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	C.DEPTID - Department	Char10	X				DeptID
2	A.NAME - Name	Char50		1			Name
3	A.EMPLID - EmplID	Char11					ID
4	B.ERNCD - Earnings Code	Char3	X	2			Earn Code
5	D.DESCR - Description	Char30					Descr
6	B.OTH_PAY - Other Pay	SNm12.2					Oth Py
7	B.EARNINGS_END_DT - Earning	Date					EarnEnd Dt
8	B.NY_ANNUAL_EARNS - NY A	Num10.2					NY Annual
9	C.NY_BARG_UNIT - Bargaining	Char2					NY Barg U
10	C.NY_EE_FTE_PCT - Employee	Num6.4					Work Pct
11	C.NY_LOC_IND - NYS Position L	Char4					Loc Ind

The screenshot shows the "Criteria" tab of the same query window. It displays a table of logical expressions:

Logical	Expression	Operator	Expression
	B.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	C.EFFDT - Effective Date	Eff Date <=	Current Dt (EffSeq = Last)
AND	C.DEPTID - Department	equal to	:1
AND	C.EMPL_STATUS - Employee St	in list	('A','L','P')
AND	B.ERNCD - Earnings Code	in list	('LOC','SDF','GEO','SED','PS1')
AND	B.OTH_PAY - Other Pay	not equal to	0
AND	(B.EARNINGS_END_DT - Earning	is null	
OR	B.EARNINGS_END_DT - Earning	not less than	:2)
AND	D.EFFDT - Effective Date	Eff Date <=	B.EFFDT - Effective Date

**00-DeductionCodePrompt**

Returns a list of employees who have a specific deduction.

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.NAME - Name	Char50		1			Name
2	A.EMPLID - EmplID	Char11					ID
3	C.DEDCD - Deduction Code	Char6					Deductn Cd
4	D.DESCR - Description	Char30					Descr
5	C.DED_ADDL_AMT - Deduction	Num11.2					Flat Amt
6	C.DED_RATE_PCT - Deduction R	Num8.3					Rate/Pct
7	C.EFFDT - Effective Date	Date					Eff Date
8	C.DEDUCTION_END_DT - Deduct	Date					Ded End

Logical	Expression	Operator	Expression
	B.EFFDT - Effective Date	Eff Date <=	Current Dt (EffSeq = Last)
AND	B.DEPTID - Department	equal to	:2
AND	D.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	D.DEDCD - Deduction Code	equal to	C.DEDCD - Deduction Code
AND	B.EMPL_STATUS - Employee St	equal to	A
AND	C.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	C.DEDCD - Deduction Code	equal to	:4
AND	(C.DED_RATE_PCT - Deduction	not equal to	0
OR	C.DED_ADDL_AMT - Deduction	greater than	0)
AND	(C.DEDUCTION_END_DT - Dedu	is null	
OR	C.DEDUCTION_END_DT - Deduct	not less than	:3)

**00-Earnings\_By\_FundCode**

Identifies the Funding Code associated with each Earnings Codes in a specific pay period.  
This is a complex query and takes 10 – 15 minutes to return results.

The screenshot shows a query window titled "PUBLIC.QUERY.00-Earnings\_By\_FundCode - Query". The left pane shows a tree view of fields: A = PAY\_CHECK - Pay Check, B = PAY\_ERN\_DIST - Pay Che, C = JOB - EE Job History, D = JOBCODE\_TBL - Job Code, E = JOBCODE\_TBL - Job Code, F = ACCT\_CD\_TBL - Account, Expressions, and Prompts. The right pane shows a table with columns: Col, Record.Field, Format, Rel, Ord, Xlt, Agg, and Heading. The table lists 20 fields from the query.

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.NAME - Name	Char50		1			Name
2	A.EMPLID - EmplID	Char11					ID
3	A.EMPL_RCD# - Employment Rc	Num3.0					Empl Rcd#
4	B.POSITION_NBR - Position Num	Char8	X				Position
5	E.DESCR - Description	Char30					JOBCODE TITLE
6	C.NY_NYS_POSITION - NYS Po	Char8					NYS Postn
7	D.DESCR - Description	Char30					NYS JOBCODE TITLE
8	C.NY_LINE_ITEM_NBR - Line Nu	Char5					Line Nbr
9	B.ERNCD - Earnings Code	Char3		2			Earn Code
10	B.EARNINGS - Earnings	SNm15.2					Earnings
11	B.POSITION_POOL_ID - Position	Char3					Pool ID
12	B.ACCT_CD - Account Code	Char25	X				Acct Code
13	F.NY_APPROP_FUND_CD - App	Char10					Approp Fun
14	B.FISCAL_YEAR - Fiscal Year	Num4.0					Year
15	B.ACCOUNTING_PERIOD - Acco	Num3.0					Period
16	B.JOBCODE - Job Code	Char6	X				Job Code
17	C.NY_NYS_JOBCODE - NYS Jo	Char6	X				NYS JobCD
18	A.PAY_END_DT - Pay Period En	Date					Pay End Dt
19	A.DEPTID - Department	Char10					DeptID
20	A.CHECK_DT - Paycheck Issue	Date					Check Dt

The screenshot shows the same query window with the "Criteria" tab selected. The left pane is the same as above. The right pane shows a table with columns: Logical, Expression, Operator, and Expression. The table lists the criteria for the query.

Logical	Expression	Operator	Expression
	A.PAY_END_DT - Pay Period En	equal to	:1
AND	A.DEPTID - Department	equal to	:2
AND	(C.EFFDT - Effective Date	Eff Date <=	A.PAY_END_DT - Pay Period End Date (EffSeq = Last)
AND	C.EMPLID - EmplID	equal to	A.EMPLID - EmplID
AND	C.EMPL_RCD# - Employment Rc	equal to	A.EMPL_RCD# - Employment Rcd Nbr)
AND	D.EFFDT - Effective Date	Eff Date <=	A.PAY_END_DT - Pay Period End Date
AND	E.EFFDT - Effective Date	Eff Date <=	A.PAY_END_DT - Pay Period End Date
AND	F.EFFDT - Effective Date	Eff Date <=	Current Dt

**00-IndividualHistory**

Returns all rows entered in the Job table for an individual employee while in your agency.

The screenshot shows a query editor window with a list of fields. The fields are organized into columns: Col, Record.Field, Format, Rel, Ord, Xlt, Agg, and Heading. The fields listed are:

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.EFFDT - Effective Date	Date		1			Eff Date
2	A.EFFSEQ - Effective Sequence	Num3.0		2			Sequence
3	A.EMPLID - EmplID	Char11	X				ID
4	A.EMPL_RCD# - Employment Rc	Num3.0					Empl Rcd#
5	B.NAME - Name	Char50					Name
6	A.DEPTID - Department	Char10	X				DeptID
7	A.NY_LINE_ITEM_NBR - Line Nu	Char5					Line Nbr
8	A.NY_BARG_UNIT - Bargaining	Char2					NY Barg U
9	A.POSITION_NBR - Position Num	Char8					Position
10	A.NY_NYS_POSITION - NYS Po	Char8					NYS Postn
11	A.ACTION - Action	Char3			N		Action
12	A.ACTION_REASON - Reason C	Char3					Reason
13	A.ACTION_DT - Action Date	Date					Action Dt
14	A.EMPL_STATUS - Employee St	Char1			N		Status
15	A.NY_FTA_SALARY - Full-Time	Num19.6					FTA Salary
16	A.JOBCODE - Job Code	Char6	X				Job Code
17	C.DESCR - Description	Char30					Title
18	A.GRADE - Salary Grade	Char3	X				Grade
19	A.FULL_PART_TIME - Full/Part Ti	Char1			N		Full/Part
20	A.FTE - Position FTE	Num4.2					FTE
21	A.NY_EE_FTE_PCT - Employee	Num6.4					Work Pct
22	A.HOURLY_RT - Hourly Rate	Num19.6					Hrly. Rate
23	A.COMPRATE - Compensation R	Num19.6					Comp Rate

The screenshot shows a query editor window with a logical expression table. The table has columns: Logical, Expression, Operator, and Expression. The expressions are:

Logical	Expression	Operator	Expression
	A.EMPLID - EmplID	equal to	:1
AND	C.EFFDT - Effective Date	Eff Date <=	A.EFFDT - Effective Date

**00-OT\_ByPayPeriod**

Returns rows for all Earnings Codes that start with the letter “O” in a specific pay period. The criteria can be modified to list specific Earnings Codes. There are some Earnings Codes that start with the letter “O” that are not overtime payments, e.g. OVP for Overpayments. Refer to page 3 for instructions on modifying Public queries.

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.PAY_END_DT - Pay Period En	Date					Pay End Dt
2	C.NAME - Name	Char50	1				Name
3	A.EMPLID - EmplID	Char11					ID
4	B.ERNCD - Earnings Code	Char3	2				Earn Code
5	A.EARNS_BEGIN_DT - Earnings	Date		3			Erns Begin
6	A.EARNS_END_DT - Earnings E	Date					Earns End
7	B.OTH_EARNS - Other Earnings	SNm12.2					Oth Earns
8	B.OTH_HRS - Other Hours	SNm8.2					Oth Hrs
9	A.DEPTID - Department	Char10	X				DeptID
10	D.NY_LINE_ITEM_NBR - Line Nu	Char5					Line Nbr

Logical	Expression	Operator	Expression
	A.DEPTID - Department	equal to	:1
AND	A.PAY_END_DT - Pay Period En	equal to	:2
AND	B.ERNCD - Earnings Code	like	O_
AND	C.EMPLID - EmplID	equal to	A.EMPLID - EmplID
AND	D.EFFDT - Effective Date	Eff Date <=	Current Dt (EffSeq = Last)
AND	A.EMPLID - EmplID	equal to	D.EMPLID - EmplID

**00-Pay\_ByMailDrop**

Returns a list of employees paid in a specific pay period. The list is sorted by the Mail Drop field. Employee who receives both a check and an advice are listed twice. Checks are identified by the name of the bank issuing the check, currently “Fleet”.

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.MAIL_DROP - Mail Drop ID	Char50		1			Mail Drop
2	B.NAME - Name	Char50		2			Name
3	A.EMPLID - EmplID	Char11	X				ID
4	A.EMPL_RCD# - Employment Rc	Num3.0		3			Empl Rcd#
5	B.DEPTID - Department	Char10					DeptID
6	B.PAY_END_DT - Pay Period En	Date					Pay End Dt
7	B.FORM_ID - Form Identification	Char6					Check or Advice

Logical	Expression	Operator	Expression
	(B.EMPLID - EmplID)	equal to	A.EMPLID - EmplID
AND	B.EMPL_RCD# - Employment Rc	equal to	A.EMPL_RCD# - Employment Rcd Nbr)
AND	B.PAY_END_DT - Pay Period En	equal to	:1
AND	B.DEPTID - Department	equal to	:2

### Archived Data Queries

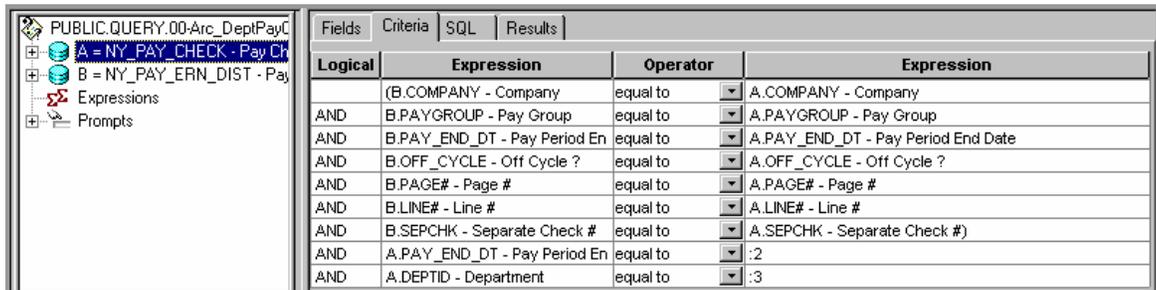
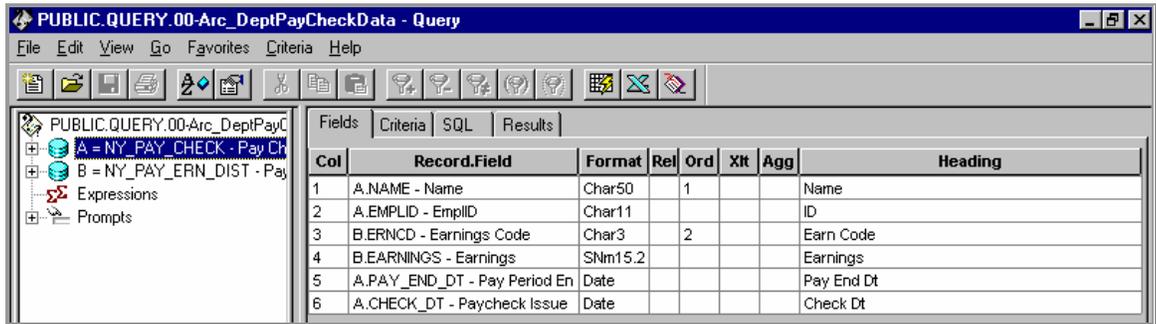
Earnings data is periodically archived or moved from the main earnings tables. The next four queries return archived earnings information.

Archived data is saved in these tables:

- Archived earnings tables:
- NY\_Pay\_Check
- NY\_Pay\_Earnings
- NY\_Pay\_Ern\_Dist
- NY\_Pay\_Oth\_Earn

#### 00-Arc\_DeptPayCheckData

Returns data from NY\_Pay\_Check and NY\_Pay\_Ern\_Dist for a single archived pay period. Includes prompts on DeptID and Pay Period End Date.



**00-Arc\_Dept\_OtherEarns**

Returns archived miscellaneous payment data (e.g. OTA) from the NY\_Pay\_Earnings and NY\_Pay\_Oth\_Earn tables. The query can be modified to include criteria for specific earnings codes. Refer to page 3 for instructions on modifying Public queries.

Col	Record.Field	Format	Rel	Ord	Xt	Agg	Heading
1	C.NAME - Name	Char50		1			Name
2	A.EMPLID - EmplID	Char11					ID
3	A.EARNS_BEGIN_DT - Earnings	Date					Earns Begin
4	A.EARNS_END_DT - Earnings E	Date					Earns End
5	B.ERNCD - Earnings Code	Char3		2			Earn Code
6	B.OTH_HRS - Other Hours	SNm8.2					Oth Hrs
7	B.OTH_EARNS - Other Earnings	SNm12.2					Oth Earns
8	A.PAY_END_DT - Pay Period En	Date					Pay End Dt

Logical	Expression	Operator	Expression
	(B.COMPANY - Company	equal to	A.COMPANY - Company
AND	B.PAYGROUP - Pay Group	equal to	A.PAYGROUP - Pay Group
AND	B.PAY_END_DT - Pay Period En	equal to	A.PAY_END_DT - Pay Period End Date
AND	B.OFF_CYCLE - Off Cycle ?	equal to	A.OFF_CYCLE - Off Cycle ?
AND	B.PAGE# - Page #	equal to	A.PAGE# - Page #
AND	B.LINE# - Line #	equal to	A.LINE# - Line #
AND	B.ADDL# - Additional Pay Line #	equal to	A.ADDL# - Additional Pay Line #)
AND	C.EMPLID - EmplID	equal to	A.EMPLID - EmplID
AND	A.DEPTID - Department	equal to	:2
AND	A.PAY_END_DT - Pay Period En	equal to	:3

**00-Arc\_Ind\_OtherEarns**

Returns a history of archived miscellaneous payments for a specific employee. Includes a prompt on EmplID.

The screenshot shows the Microsoft Access interface for a query named 'PUBLIC.QUERY.00-Arc\_Ind\_OtherEarns - Query'. The 'Fields' tab is selected, displaying a list of fields from three tables: A = NY\_PAY\_EARNINGS - Pa, B = NY\_PAY\_OTH\_EARN - P, and C = PERSONAL\_DATA - EE P. The fields are listed in a table with columns for Col, Record.Field, Format, Rel, Ord, XR, Agg, and Heading.

Col	Record.Field	Format	Rel	Ord	XR	Agg	Heading
1	A.PAY_END_DT - Pay Period En	Date		1			Pay End Dt
2	C.NAME - Name	Char50					Name
3	A.EMPLID - EmplID	Char11					ID
4	A.EARNS_BEGIN_DT - Earnings	Date					Erns Begin
5	A.EARNS_END_DT - Earnings E	Date					Earns End
6	B.ERNCD - Earnings Code	Char3		2			Earn Code
7	B.OTH_EARNS - Other Earnings	SNm12.2					Oth Earns
8	B.OTH_HRS - Other Hours	SNm8.2					Oth Hrs

The screenshot shows the 'Criteria' tab in Microsoft Access. It displays a table with columns for Logical, Expression, Operator, and Expression. The criteria are defined for fields from tables A, B, and C.

Logical	Expression	Operator	Expression
	B.COMPANY - Company	equal to	A.COMPANY - Company
AND	B.PAYGROUP - Pay Group	equal to	A.PAYGROUP - Pay Group
AND	B.PAY_END_DT - Pay Period En	equal to	A.PAY_END_DT - Pay Period End Date
AND	B.OFF_CYCLE - Off Cycle ?	equal to	A.OFF_CYCLE - Off Cycle ?
AND	B.PAGE# - Page #	equal to	A.PAGE# - Page #
AND	B.LINE# - Line #	equal to	A.LINE# - Line #
AND	B.ADDL# - Additional Pay Line #	equal to	A.ADDL# - Additional Pay Line #
AND	A.EMPLID - EmplID	equal to	:1
AND	C.EMPLID - EmplID	equal to	A.EMPLID - EmplID

**00-Arc\_Ind\_PayCheckData**

Returns a history of archived earnings information for a specific employee. Includes a prompt on EmplID.

The screenshot shows a query window with a toolbar and a field list on the left. The main area displays a table with the following columns: Col, Record.Field, Format, Rel, Ord, Xlt, Agg, and Heading. The data rows are as follows:

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.CHECK_DT - Paycheck Issue	Date					Check Dt
2	A.NAME - Name	Char50					Name
3	A.EMPLID - EmplID	Char11					ID
4	B.ERNCD - Earnings Code	Char3		2			Earn Code
5	B.EARNINGS - Earnings	SNm15.2					Earnings
6	A.PAY_END_DT - Pay Period En	Date		1			Pay End Dt

The screenshot shows the same query window with the 'Criteria' tab selected. The main area displays a table with the following columns: Logical, Expression, Operator, and Expression. The data rows are as follows:

Logical	Expression	Operator	Expression
	B.COMPANY - Company	equal to	A.COMPANY - Company
AND	B.PAYGROUP - Pay Group	equal to	A.PAYGROUP - Pay Group
AND	B.PAY_END_DT - Pay Period En	equal to	A.PAY_END_DT - Pay Period End Date
AND	B.OFF_CYCLE - Off Cycle ?	equal to	A.OFF_CYCLE - Off Cycle ?
AND	B.PAGE# - Page #	equal to	A.PAGE# - Page #
AND	B.LINE# - Line #	equal to	A.LINE# - Line #
AND	B.SEPCHK - Separate Check #	equal to	A.SEPCHK - Separate Check #
AND	A.EMPLID - EmplID	equal to	:1

“Difference” Queries

The following two queries return rows where there is different information in two tables. This is not necessarily incorrect but should be reviewed to determine if changes are needed. Some employees prefer different addresses or names in these two tables.

00-DifferentAddresses

Returns rows where the employee’s address in the Personal Data table is different from his/her address in the Pay\_Check table.

The screenshot shows a query window with the following field list:

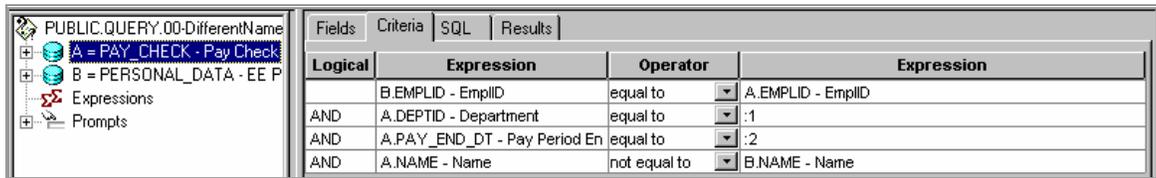
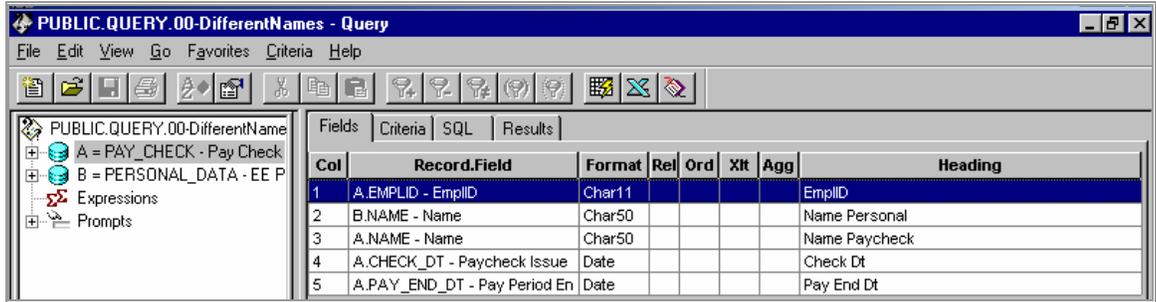
Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.NAME - Name	Char50		1			Name
2	A.ADDRESS1 - Address Line 1	Char35					PayCheck Address
3	A.ADDRESS2 - Address Line 2	Char35					PayCheck Address 2
4	A.CITY - City	Char30					PayCheck City
5	A.STATE - State	Char6					PayCheck State
6	A.ZIP - Postal Code	Char10					PayCheck Zip
7	B.ADDRESS1 - Address Line 1	Char35					PersonalData Address
8	B.ADDRESS2 - Address Line 2	Char35					PersonalData Address 2
9	B.CITY - City	Char30					PersonalData City
10	B.STATE - State	Char6					PersonalData State
11	B.ZIP - Postal Code	Char10					PersonalData Zip
12	A.PAY_END_DT - Pay Period En	Date					Pay End Dt
13	A.CHECK_DT - Paycheck Issue	Date					Check Dt
14	A.EMPLID - EmplID	Char11					ID

The screenshot shows the SQL criteria for the query:

Logical	Expression	Operator	Expression
	B.EMPLID - EmplID	equal to	A.EMPLID - EmplID
AND	A.DEPTID - Department	equal to	:1
AND	A.PAY_END_DT - Pay Period En	equal to	:2
AND	(A.ADDRESS1 - Address Line 1	not equal to	B.ADDRESS1 - Address Line 1
OR	A.CITY - City	not equal to	B.CITY - City
OR	A.ZIP - Postal Code	not equal to	B.ZIP - Postal Code)

**00-DifferentNames**

Returns rows where the employee’s name in the Pay\_Check table is different from his/her name in the Personal Data table. This is not necessarily incorrect. Some employees prefer different names in these two tables.



### Duplicate Overtime Payment Queries

The next three queries identify duplicate overtime payments. The purpose is to determine if multiple overtime rows have been entered for the same employee for the same day(s).

#### 00-Dup\_OT\_Audit\_Daily

Identifies when multiple overtime rows have been entered for the same employee for the same day(s). Contains a prompt on the data entry date. Run the 00-Dup\_OT\_Verification query to verify the results.

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.DEPTID - Department	Char10		1			DeptID
2	A.EMPLID - EmplID	Char11		2			ID
3	D.NAME - Name	Char50					Name
4	A.EARNS_BEGIN_DT - Earnings	Date		3			Erns Begin Date
5	A.EARNS_END_DT - Earnings E	Date					Earns End Date
6	A.PAY_END_DT - Pay Period En	Date					Pay End Dt
7	A.ERNCD - Earnings Code	Char3		4			Earn Code
8	A.NY_TIME_COMMENTS - NY Ti	Char254					Comments
9	A.OPRID - Operator Id	Char8					Entered By
10	A.NY_HOURS - Hours	SNm8.2					Hours
11	A.DATE_TIME - Date Time	DateTm					Date/Time of Entry

Logical	Expression	Operator	Expression
	A.DEPTID - Department	equal to	:1
AND	A.DATE_TIME - Date Time	greater than	:3
AND	A.ERNCD - Earnings Code	like	O_
AND	A.EMPLID - EmplID	in list	Subquery
AND	D.EMPLID - EmplID	equal to	A.EMPLID - EmplID

#### Associated Subquery

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	B.EMPLID - EmplID	Char11					ID

Logical	Expression	Operator	Expression
	B.PAY_END_DT - Pay Period En	not equal to	A.PAY_END_DT - Pay Period End Date
AND	(B.EMPLID - EmplID)	equal to	A.EMPLID - EmplID
AND	B.EMPL_RCD# - Employment Rc	equal to	A.EMPL_RCD# - Employment Rod Nbr
AND	B.EARNS_BEGIN_DT - Earnings	equal to	A.EARNS_BEGIN_DT - Earnings Begin Date
AND	B.EARNS_END_DT - Earnings E	equal to	A.EARNS_END_DT - Earnings End Date
AND	C.ERNCD - Earnings Code	equal to	A.ERNCD - Earnings Code)

**00-Dup\_OT\_Audit**

Similar to previous query but contains a prompt on the pay period end date, instead of the data entry date.

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.DEPTID - Department	Char10		1			Agency
2	D.NAME - Name	Char50					Name
3	A.EMPLID - EmplID	Char11		2			SS#
4	A.EARNS_BEGIN_DT - Earnings	Date		3			Erns Begin Date
5	A.EARNS_END_DT - Earnings E	Date					Earns End Date
6	A.PAY_END_DT - Pay Period En	Date					Pay End Dt
7	A.ERNCD - Earnings Code	Char3		4			Earn Code
8	A.NY_TIME_COMMENTS - NY Ti	Char254					Time Entry Comments
9	A.OPRID - Operator Id	Char8					Entered By
10	A.NY_HOURS - Hours	SNm8.2					Hours
11	A.NY_AMOUNT - Amount	SNm12.2					Amount
12	A.NY_DAYS - NY Days	SNm7.2					Days
13	A.NY_UNITS - New York Units	SNm4.0					Units
14	A.DATE_TIME - Date Time	DateTm					Date/Time of Entry

Logical	Expression	Operator	Expression
	A.DEPTID - Department	equal to	:1
AND	A.PAY_END_DT - Pay Period En	equal to	:2
AND	A.ERNCD - Earnings Code	like	O_
AND	A.EMPLID - EmplID	in list	Subquery
AND	D.EMPLID - EmplID	equal to	A.EMPLID - EmplID

**Associated Subquery**

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	B.EMPLID - EmplID	Char11					ID

Logical	Expression	Operator	Expression
	B.PAY_END_DT - Pay Period En	not equal to	A.PAY_END_DT - Pay Period End Date
AND	(B.EMPLID - EmplID	equal to	A.EMPLID - EmplID
AND	B.EMPL_RCD# - Employment Rcd	equal to	A.EMPL_RCD# - Employment Rcd Nbr
AND	B.EARNS_BEGIN_DT - Earnings	equal to	A.EARNS_BEGIN_DT - Earnings Begin Date
AND	B.EARNS_END_DT - Earnings E	equal to	A.EARNS_END_DT - Earnings End Date
AND	C.ERNCD - Earnings Code	equal to	A.ERNCD - Earnings Code)

**00-Dup\_OT\_Verification**

The purpose of this query is to verify if a specific employee listed in either of the two prior queries has received a previous payment for the same days of overtime that are being processed in the current pay period. Review the history of overtime payments since 01/01/01 made to a specific employee to determine if the duplicated OT payments need to be corrected.

The screenshot shows a query window titled 'PUBLIC.QUERY.00-Dup\_OT\_Verification - Query'. The 'Fields' tab is active, displaying a list of fields from two tables: 'A = PAY\_EARNINGS - Payroll' and 'B = PAY\_OTH\_EARNS - Payroll'. The fields are organized into columns: Col, Record, Field, Format, Rel, Ord, Xit, Agg, and Heading.

Col	Record	Field	Format	Rel	Ord	Xit	Agg	Heading
1	A	PAY_END_DT - Pay Period En	Date					Pay End Dt
2	A	EMPLID - EmplID	Char11					ID
3	A	EMPL_RCD# - Employment Rc	Num3.0					Empl Rcd#
4	A	EARN_BEGIN_DT - Earnings	Date		1			Erns Begin
5	A	EARN_END_DT - Earnings E	Date					Earns End
6	B	ERNCD - Earnings Code	Char3		2			Earn Code
7	B	OTH_EARNS - Other Earnings	SNm12.2					Oth Earns

The screenshot shows the 'Criteria' tab of the same query window. It displays logical expressions for the query. The columns are Logical, Expression, Operator, and Expression.

Logical	Expression	Operator	Expression
	A.EMPLID - EmplID	equal to	:1
AND	B.ERNCD - Earnings Code	like	O_
AND	A.PAY_END_DT - Pay Period En	greater than	2001-01-01

### “Position” Queries

#### 00-PositionPool\_Funding

Returns a list of agency Position Pools and associated funding information.

The screenshot shows a query window titled "PUBLIC.QUERY.00-PositionPool\_Funding - Query". The interface includes a menu bar (File, Edit, View, Go, Favorites, Criteria, Help) and a toolbar with various icons. On the left, a tree view shows the query structure with tables "A = POOL\_TBL - Pool Definitio" and "B = ACCT\_CD\_TBL - Account". The main area displays a table with the following columns: Col, Record.Field, Format, Rel, Ord, Xlt, Agg, and Heading. The data rows are:

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.POSITION_POOL_ID - Position	Char3		1			Pool ID
2	A.ACCT_CD - Account Code	Char25	X	2			Account Code
3	B.NY_SEGR_PROG_CD - Segr P	Char10					Segregate
4	B.NY_SEGR_AUTH_CD - Segr	Char3					Segregate
5	B.DESCR - Description	Char30					Description

The screenshot shows the same query window with the "Criteria" tab selected. It displays a table with the following columns: Logical, Expression, Operator, and Expression. The criteria are:

Logical	Expression	Operator	Expression
	A.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	B.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	A.DEPTID - Department	equal to	:1

**00-Positions\_FilledActive**

Returns a list of currently filled positions in the agency. It includes data from the Job table associated with the employee currently filling the position.

**PUBLIC.QUERY.00-PositionsFilledActive - Query**

File Edit View Go Favorites Criteria Help

Fields Criteria SQL Results

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.POSITION_NBR - Position Num	Char8		1			Position
2	A.DESCR - Description	Char30					Descr
3	B.EMPLID - EmplID	Char11	X				ID
4	C.NAME - Name	Char50					Name
5	A.NY_LINE_ITEM_NBR - Line Nu	Char5					Line Nbr
6	A.REG_TEMP - Regular/Tempora	Char1				N	Reg/Temp
7	A.FULL_PART_TIME - Full/Part Ti	Char1				N	Full/Part
8	A.FTE - Position FTE	Num4.2					FTE
9	A.POSITION_POOL_ID - Position	Char3	X				Pool ID
10	A.GRADE - Salary Grade	Char3	X				Grade
11	A.NY_PAY_BASIS_CD - Pay Ba	Char3					Pay Basis
12	A.NY_BARG_UNIT - Bargaining	Char2					NY Barg U
13	A.BUDGETED_POSN - Budgeted	Char1					Budgeted
14	A.EMPL_TYPE - Employee Type	Char1				N	Type
15	B.EMPL_STATUS - Employee St	Char1				S	Employee Status

**PUBLIC.QUERY.00-PositionsFilledActive**

Fields Criteria SQL Results

Logical	Expression	Operator	Expression
	A.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	A.EFF_STATUS - Status as of E	equal to	A
AND	A.DEPTID - Department	equal to	:1
AND	B.EFFDT - Effective Date	Eff Date <=	Current Dt (EffSeq = Last)
AND	(B.DEPTID - Department	equal to	A.DEPTID - Department
AND	B.POSITION_NBR - Position Num	equal to	A.POSITION_NBR - Position Number)
AND	B.EMPL_STATUS - Employee St	in list	('A','L','P')

**00-Positions\_VacantActive**

Returns a list of currently vacant active positions in the agency.

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	A.POSITION_POOL_ID - Position	Char3	X	1			Pool ID
2	A.POSITION_NBR - Position Num	Char8		2			Position
3	A.NY_LINE_ITEM_NBR - Line Nu	Char5					Line Nbr
4	A.DESCR - Description	Char30					Descr
5	A.POSN_STATUS - Position Stat	Char1				S	Position Status
6	A.GRADE - Salary Grade	Char3	X				Grade
7	A.NY_BARG_UNIT - Bargaining	Char2					NY Barg U
8	A.EFF_STATUS - Status as of E	Char1				S	Status
9	A.DEPTID - Department	Char10	X				Department

Logical	Expression	Operator	Expression
	A.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	A.DEPTID - Department	equal to	:1
AND	A.POSITION_NBR - Position Num	not in list	Subquery
AND	A.EFF_STATUS - Status as of E	equal to	A

**Associated Subquery**

Col	Record.Field	Format	Rel	Ord	Xlt	Agg	Heading
1	B.POSITION_NBR - Position Num	Char8					Position

Logical	Expression	Operator	Expression
	B.EFFDT - Effective Date	Eff Date <=	Current Dt (EffSeq = Last)
AND	B.DEPTID - Department	equal to	A.DEPTID - Department
AND	B.EMPL_STATUS - Employee St	equal to	A

**00-PS\_Certificate**

Returns same data as the Reveal NHRP723 (PS Certificates) report. If run to Excel, the data can then be sorted and manipulated as the agency chooses.

Col	Record.Field	Format	Rel	Ord	Xit	Agg	Heading
1	A.DEPTID - Department	Char10	X	1			DeptID
2	C.NY_APPROP_FUND_CD - App	Char10		2			Fund/Acct
3	C.NY_APPROP_PROG_CD - Ap	Char10		3			Program
4	C.NY_COST_CENTER - Cost Ce	Char6		4			Cost Center
5	C.NY_EXPENDITURE_VAR - Var	Char2		5			Variable
6	C.BUDGET_YEAR - Budget Yea	Char4					Budget Year
7	A.POSITION_POOL_ID - Position	Char3	X	6			Pool ID
8	B.DESCR - Description	Char30					Pool Descr
9	A.NY_LINE_ITEM_NBR - Line Nu	Char5		7			Line Nbr
10	A.POSITION_NBR - Position Num	Char8					Position
11	A.DESCR - Description	Char30					Descr
12	A.GRADE - Salary Grade	Char3	X				Grade
13	A.REG_TEMP - Regular/Tempora	Char1				S	Reg/Temp/Seas
14	A.NY_POS_EXP_DT - Position E	Date					Pos End Dt
15	A.FTE - Position FTE	Num4.2					Position FTE
16	A.NY_APPROVE_SAL_RT - Ap	Num19.6					Not To Exceed Salary
17	A.EFFDT - Effective Date	Date					Position Effective Date

Logical	Expression	Operator	Expression
	A.DEPTID - Department	equal to	:1
AND	A.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	C.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	B.EFFDT - Effective Date	Eff Date <=	Current Dt
AND	A.EFF_STATUS - Status as of E	equal to	A