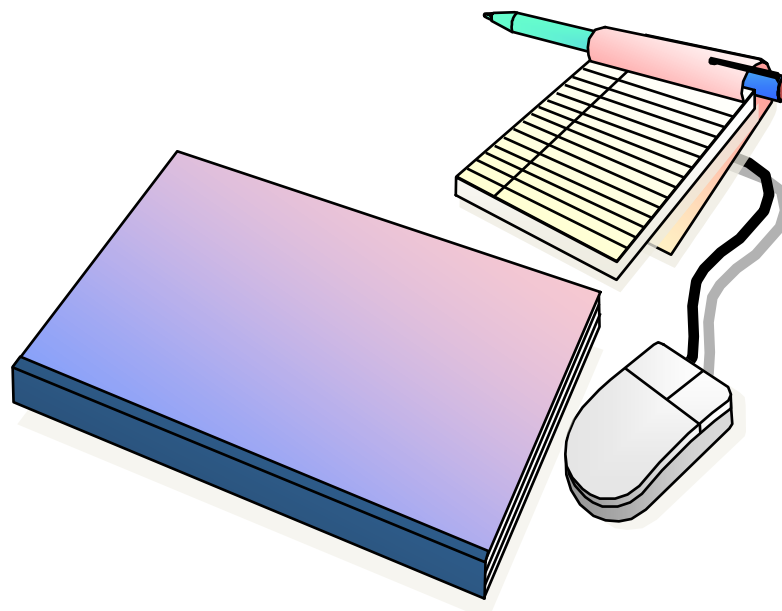


Ryska Fartyg



Innehållsförteckning

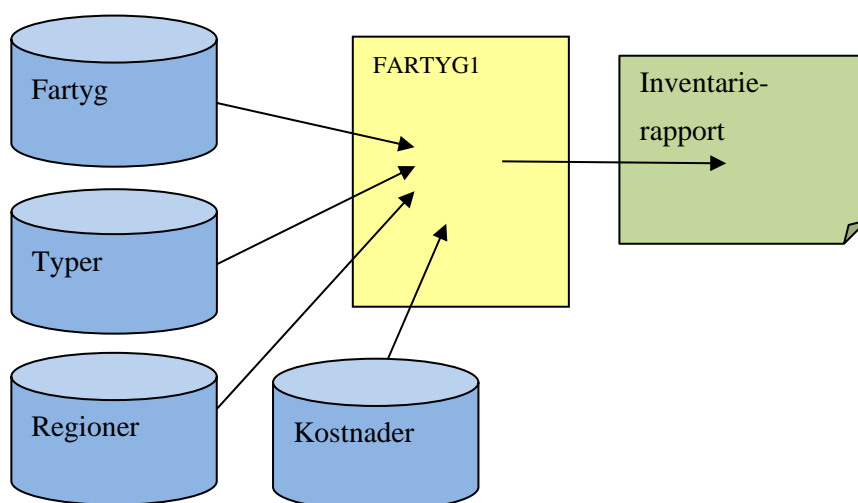
Ryska Fartyg	3
Uppgift 1 – Filer	3
Uppgift 2 – Inventarielista	4
Uppgift 3 – Inventarielista, begränsningar.....	5
Uppgift 4 – Kostnadsökning	5
Uppgift 5 – Utbackning, JCL.....	6
Uppgift 6 – JCL-procedur.....	7
Uppgift 7 – Sökning av fartyg	8

Ryska Fartyg

Ryska marinen har ett antal filer som beskriver detaljer om olika fartyg i olika regioner av världen. Programmet FARTYG kommer att göra en sammanställning över dessa fartyg där rapporten visar vilka fartyg som finns i olika regioner, bruttovikt, besättningsstorlek och månadskostnader.

Uppgift 1 – Filer

- Översiktlig bild av filer och program.



- **Fartyg-filen**, beskriver information om varje fartyg som patrullerar i olika regioner och har följande utseende:

Fält	Typ	Längd	Innehåll
Fartygs-Namn	X	20	--
Fartygs-Typ	9	2	01-12
Bruttovikt	9	6	0-999999
Besättning	9	5	0-99999
Region	X	1	1/2/3/4/5

- Ladda ned innehållet i denna fil från cobolskolan.se/CA01 under rubriken *Ryska Fartyg* där filen har namnet *Fartyg-Info*. Placera innehållet i biblioteket `userid.CA01.COBDATA` och ge medlemmen namnet `FARTINFO`.
- **Typer-filen**, som beskriver information om olika fartygstyper, har följande utseende:

Fält	Typ	Längd	Innehåll
Typ	9	2	01-99
Funktion	X	25	--

Ladda ned innehållet i denna fil från cobolskolan-se/CA01 under rubriken *Ryska Fartyg* där filen har namnet *Fartyg-Typer*. Placera innehållet i biblioteket `userid.CA01.COBDATA` och ge medlemmen namnet `FARTTYP`.

- Filen med **Regioner**, som innehåller namn på regioner, där fartygen just nu opererar, har följande utseende:

Fält	Typ	Längd	Innehåll
Regionskod	X	1	1-9
Regionsnamn	X	20	--

- Ladda ned innehållet i denna fil från cobolskolan-se/CA01 under rubriken *Ryska Fartyg* där filen har namnet *Fartyg-Regioner*. Placera innehållet i biblioteket `userid.CA01.COBDATA` och ge medlemmen namnet `FARTREG`.
- Skapa en RRDS-fil och fyll med datat. Regionskod är Relativ-Record-Nummer (RRN).
- Filens namn kan vara `userid.CA01.FARTREG.RRDS`.
- Kontrollera resultatet i File Manager.

- **Kostnader-filen** har följande utseende

Fält	Typ	Längd	Innehåll
Regionskod	X	48	Kostnad per månad, 12 ggr

- Filen innehåller en rad för varje region (f n 5), där varje rad innehåller månadskostnaden (4 siffror, Rubel) för en besättningsmedlem för varje fartygstyp (f.n 12).
- Ladda ned innehållet i denna fil från cobolskolan-se/CA01 under rubriken *Ryska Fartyg* där filen har namnet *Fartyg-Kostnader*. Placera innehållet i biblioteket `userid.CA01.COBDATA` och ge medlemmen namnet `FARTKOST`.

Uppgiften är avslutad.

Uppgift 2 – Inventarielista

- Programmet `FARTYG1` skall skiva en inventarielista över fartyg som tillhör den ryska marinen. Listan skall sorteras i *Regionsordning* och *Fartygstyp*. Sorteringen av filen `FARTINFO` skall ske med en COBOL-Internsortering.
- Kostnader-filen skall läsas in i en tabell i programmet, precis som *Typer*-filen.

- Rapporten kan ha följande utseende:

```
USSR Inventarielista
Region  Fartygs-typ  Fartygs-namn  Bruttovikt  Besättning  Månadskostnad
. . . . .      . . . . .      . . . . .      . . . . .      . . . . .

< https://cobolskolan.se/wp-content/uploads/2019/11/Fartyg1-Rapport.txt>

*** Slut på Rapport ***
```

- Kompilera och korrigera tills du har en felfri kompilering.
- Exekvera programmet och kontrollera resultatet.

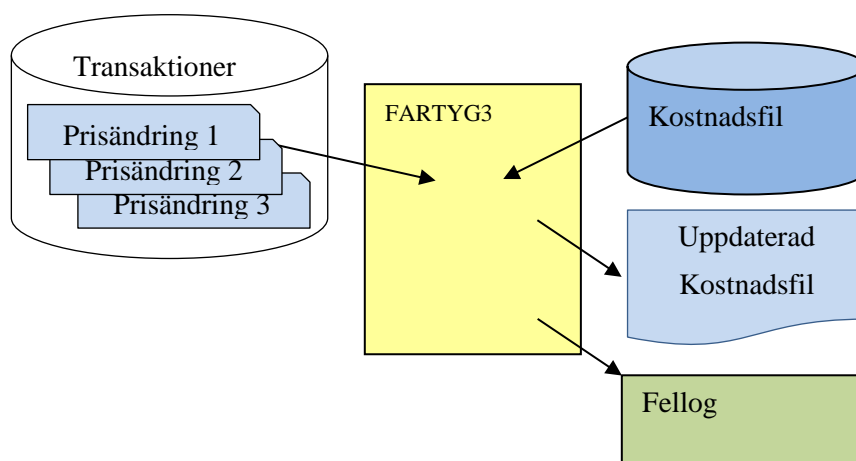
Uppgiften är avslutad.

Uppgift 3 – Inventarielista, begränsningar

- Använd programmet FARTYG1 som mall till denna uppgift. Programmet skall ha namnet FARTYG2
- Programmet skall bara skriva information om ubåtar samt fartyg med bruttovikt större än 3.500 ton.

Uppgift 4 – Kostnadsökning

- Ryska marinen har aviserat lönehöjningar. Höjningarna påverkar vissa fartygstyper i Stilla Havet och Medelhavet.



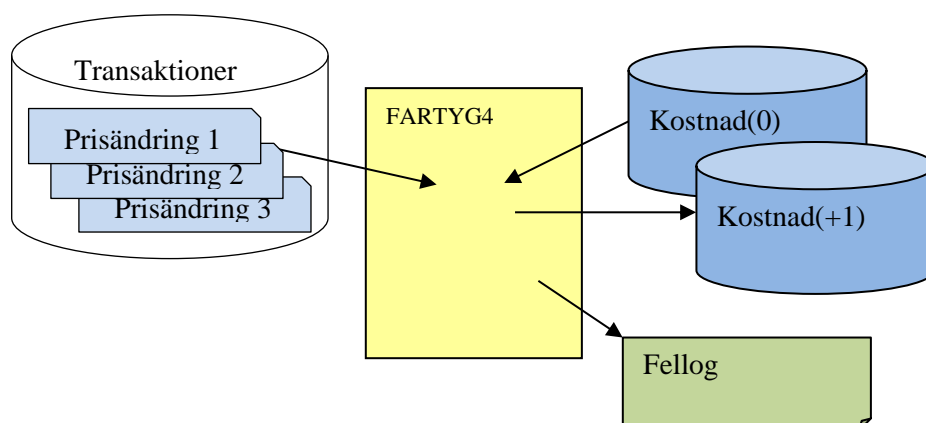
- Ändringbeställningarna är i form av transaktioner som skall läsas in från en fil och har följande utseende

Fält	Typ	Längd	Innehåll
Regionsnummer	9	2	00-99
Blank	x	1	--
Båttyp	9	2	00-99
Blank	x	1	--
Procent	9	2	01-99

- Programmet FARTYG3 skall läsa transaktioner från FARTTRAN. Bestäm själv vilka transaktioner Du vill ha, men gör avsiktligt någon/några felaktiga.
- Transaktionerna kan placeras som Instream-Data eller på en egen fil med namnet *userid.CA01.FARTRAN*.
- Bestäm själv vilka fartygstyper som berörs i de två regionerna.
- Kontrollera att transaktionens information är korrekt med avseende på regionsnummer och fartygstyp.
- Den uppdaterade kostnadsfilen kan Du skriva till Sysout i denna uppgiften. Du kommer att ändra detta i nästa uppgift.
- Felaktiga transaktioner skall skrivas ut på en fellogg. Bestäm själv formatet.
- Kompilera och korrigerar tills du har en felfri kompilering.
- Exekvera programmet och kontrollera resultatet.

Uppgiften är avslutad.

Uppgift 5 – Utbackning, JCL



- Programmet FARTYG4 (kopia av FARTYG3) skall som tidigare läsa, analysera och utföra samtliga transaktioner, MEN om det har upptäckts någon felaktig transaktion så skall programmet avslutas med retur-kod 66 i stället för noll (0) som är underförstått värde.
- Komplettera programmet.

- Du skall skapa en *Generation Data Group*, där kostnadsfilen ingår.
- Namnet på GDG skall vara *userid.CA01.FARTYG.KOSTNAD.GDG*.
- Ändra din exekverings-jJCL för programmet FARTYG4, så att en ny generation av Kostnad (+1) skapas när du kör programmet.
- Om programmet har signalerat felaktiga transaktioner (RC=66) så skall jobbet se till att den nya generationen raderas automatiskt.
Skicka ditt jobb för exekvering oh kontrollera resultatet.
- Korrigera dina transaktioner så att endast korrekta transaktioner finns med.
- Skicka jobbet för exekvering och kontrollera resultatet

Uppgiften är avslutad.

Uppgift 6 – JCL-procedur

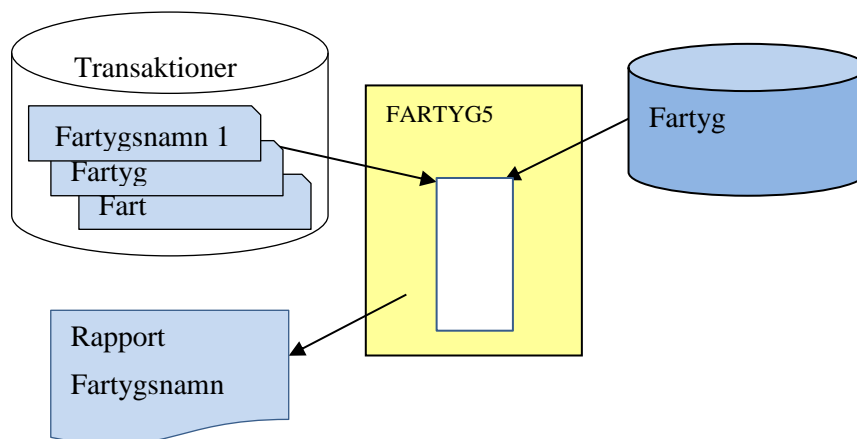
- Du skall göra en procedur av ditt exekveringsjobb för programmet FARTYG2. Proceduren kan ha namnet FARTPROC.
- Proceduren skall kunna ange en parameter för att med denna referera tidigare generationer (.Kostnad.GDG(-n)) av kostnadsfilen.
- Om parameter EJ anges, så skall generationsnummer noll (0) vara et underförstått värde.

För att försäkra dig om att det finns några tidigare generationer av kostnadsfilen, så kör du jobbet för prisändring några gånger, med olika korrekta transaktioner.

- Kör sedan proceduren FARTPROC och peka ut en tidigare generation av kostnadsfilen.
- Kontrollera resultatet.
- Kör proceduren igen med en ännu tidigare generation av kostnadsfilen. Kontrollera resultatet och jämför med den tidigare körningen.

Uppgiften är avslutad

Uppgift 7 – Sökning av fartyg



- Programmet FARTYG5 skall läsa poster med fartygsnamn.
- Fartygsnamnet skall användas för att söka information om fartyget.
- Om ej hela fartygsnamnet har angetts så skall samtliga fartyg visas som uppfyller sökkravet, t.ex. om FAR anges, så skall FARTYG1, FARTYG2 etc. visas.
- Söknamnet skall då visas i Status-fältet med en avslutande asterisk (*), vilket i ovanstående exempel skulle bli FAR*.
- Vid misslyckad sökning så skall 'I/U' visas i statusfältet, annars 'Ok'.
- För varje funnet fartyg skall en rad skrivas med fartygsinformation samt i vilken region fartyget befinner sig samt även bruttovikt och besättningsantal.
- Samtliga fartygsposter skall läsas in i en intern COBOL-tabell för att sedan användas för slumpvis sökning.
- Antalet fartyg i Fartyg-filen är okänt.

Rapporten kan se ut på detta sätt:

```

*** Fartygssökning ***
Fartygsnamn Status Typ Region Vikt Personal

```

- Kompilera och korrigera tills du har en felfri kompilering.
- Exekvera programmet och kontrollera resultatet.

Uppgiften är avslutad.

Övningen är avslutad.