

## Insert This 4.0

Ett program av Anders Persson, till hjälp för att fylla i text samt automatisera repetitiva arbetsuppgifter vid datorn.

### Grunder (Infogning av text)

Öppna och göm Insert-This-fönstret med Ctrl+Shift eller genom att klicka på programmets ikon i aktivitetsfältet. (Om du stänger fönstret på "krysset" så avslutas programmet).

Insert-This-fönstret består till största delen av en texteditor. Där skriver du (eller klistrar in) det som du vill kunna infoga i andra program.

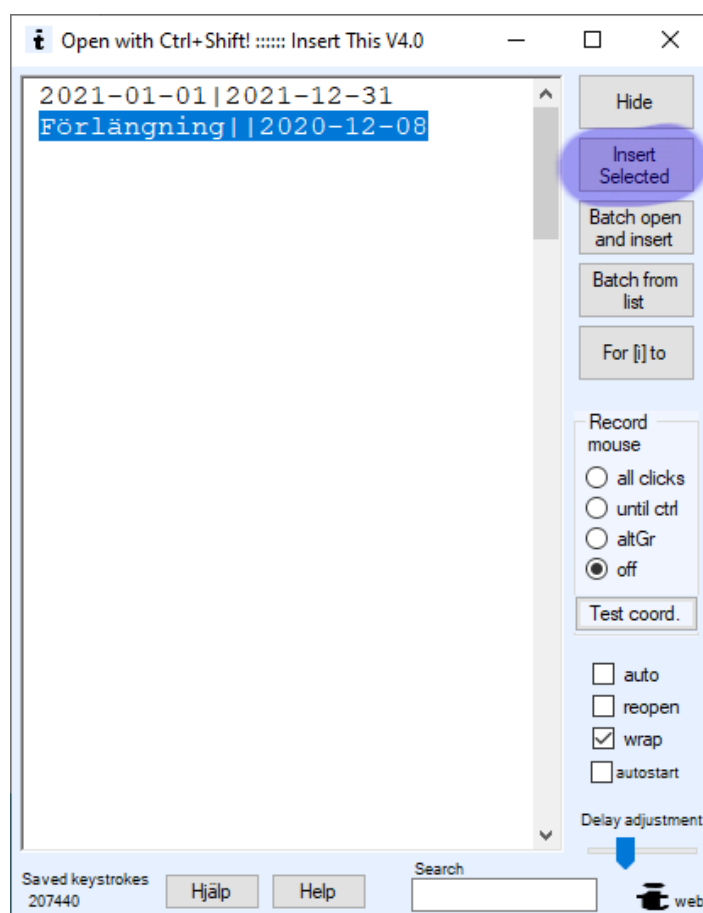
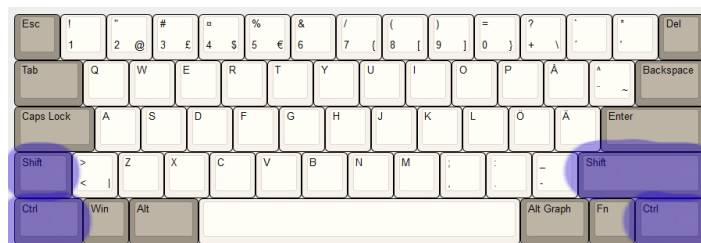
När Insert-This-fönstret är öppet och aktivt så kan du infoga markerad text i ett annat program genom att trycka på Enter (eller *AltGr+Shift*) eller med knappen "Insert Selected".

Fönstret kommer då att gömmas och texten kommer att skrivas in i det program och fält där du befann dig innan du öppnade Insert-This-fönstret.

*(AltGr+Shift infogar även när Insert-This-fönstret är gömt)*

Exempel 1 (ett sätt):

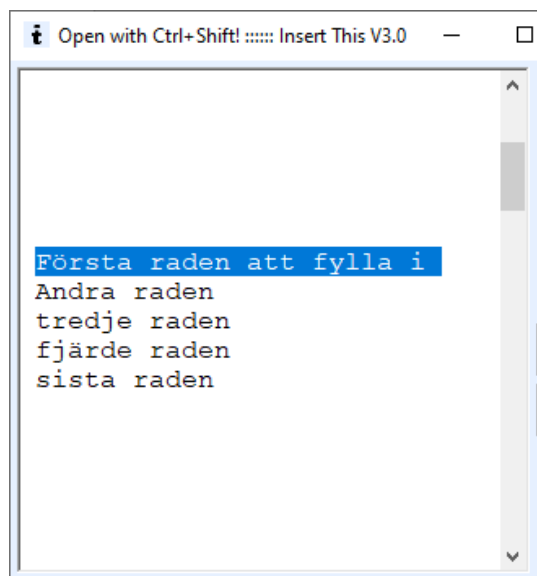
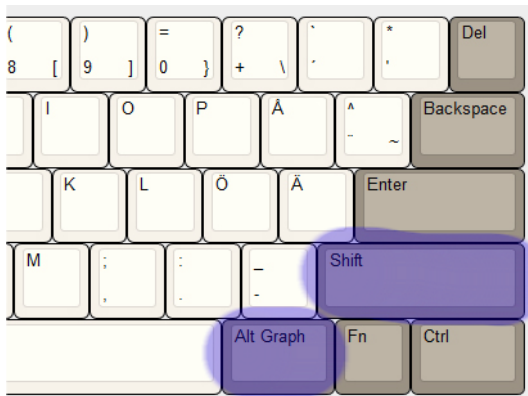
- Göm Insert-This-fönstret med Ctrl+Shift.
- Gå till det program och fält där du vill fylla i något.
- Öppna "Insert This" med Ctrl+Shift.
- Markera det du vill infoga, till exempel med piltangenterna upp och ner (som markerar en hel rad åt gången)
- Tryck sedan på Enter.



Exempel 2 (ett annat sätt):

- Öppna "Insert This" med Ctrl+Shift.
- Markera det du vill infoga.
- Gå till det program och fält där du vill fylla i något.
- Tryck på AltGr+Shift

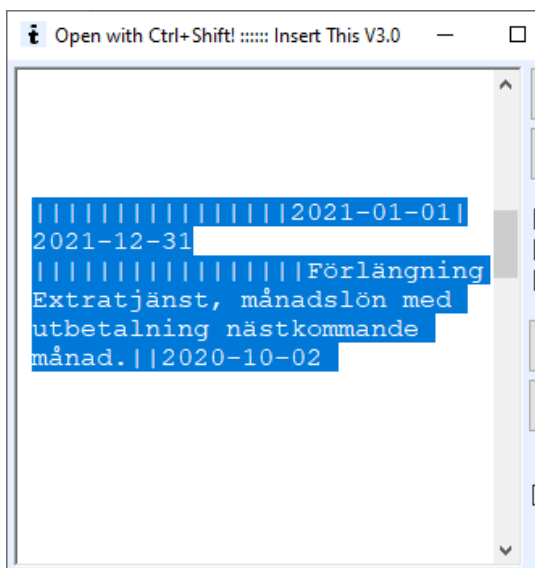
## Att fylla i flera fält i rätt ordning, metod 1



När fönstret är gömt så kan du använda AltGr+Shift för att infoga nästa rad **utan att öppna fönstret**. Varje gång du gör detta så tar den automatiskt nästa rad i Insert-this-fönstret utan att öppna det. (Men om nästa rad är tom så stannar den på den sista raden med innehåll).

På så sätt så kan du göra en lista i Insert-this-fönstret med det som skall infogas i rätt ordning och sedan bara använda AltGr+Shift i fält efter fält för att infoga raderna i rätt ordning.

När du sedan åter öppnar fönstret så är den först infogade raden åter markerad.



## Metod 2: Automatiska tabbar

Om du skriver |-streck i texten så kommer dessa att översättas till tab-tangenten. Man kan på så sätt fylla i flera fält med en enda knapptryckning, t.ex fält1|fält2|fält3

Det går också att använda teckenkoderna {TAB}, t.ex {TAB 12} betyder 12 stycken tabbar. (Se avsnittet "teckenkoder").

Om det som skall infogas är ensamt på en rad med en tom rad under så kan du hela tiden använda AltGr+Shift för att infoga samma sak gång på gång utan att behöva öppna insert-this-fönstret.

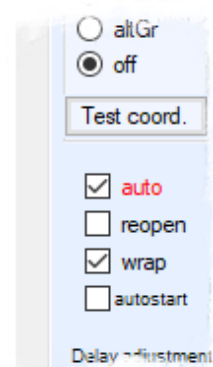
I exemplet ovan så är markeringen bara en enda rad, men eftersom "Wrap" är ikryssat så bryts raden i fönstret så att man skall se hela raden.

Man kan använda markörtangenterna för att gå upp och ner och markera en hel rad.

Man kan använda Shift och markörtangenterna för att markera flera rader samtidigt eller så kan man använda musen till det.

Home och End hoppar till början respektive slutet på texten.

Om Auto är ikryssat så infogas markerad text så fort den blir markerad, antingen med musen eller med Shift och markörtangenterna. När du släpper upp shift så infogas det. Man kan även med musen markera en hel rad genom att klicka framför starten på raden. OBS: Som en försiktighetsåtgärd så stängs auto av automatiskt så fort du editerar texten i Insert-this-fönstret.



Re-open: Om alternativet är valt så öppnas fönstret på nytt direkt efter att något infogats. Detta är nog mest användbart om man bara vill använda musen.

Wrap: Påverkar hur texten visas i fönstret. Långa rader bryts så att hela raden syns.

**Detta dokument** finns (till stor del) tillgängligt om du klickar på "Hjälp" i Insert-this-fönstret, men då enbart text utan bilder. Om du klickar på "Help" så får du istället en engelsk hjälptext.

## Teckenkoder

I texten kan man använda ett antal koder som Microsoft har implementerat. Följande beskrivning är kopierad från deras dokumentation.

*The plus sign (+), caret (^), percent sign (%), tilde (~), and parentheses () have special meanings. To specify one of these characters, enclose it within braces {}. For example, to specify the plus sign, use "{+}". To specify brace characters, use "{{}" and "}". Brackets ([ ]) have no special meaning, but you must enclose them in braces.*

*To specify characters that aren't displayed when you press a key, such as ENTER or TAB, and keys that represent actions rather than characters, use the codes in the following table.*

**Table 1**

<b>Key</b>	<b>Code</b>
BACKSPACE	{BACKSPACE}, {BS}, or {BKSP}
BREAK	{BREAK}
CAPS LOCK	{CAPSLOCK}
DEL or DELETE	{DELETE} or {DEL}
DOWN ARROW	{DOWN}
END	{END}
ENTER	{ENTER} or ~
ESC	{ESC}
HELP	{HELP}
HOME	{HOME}
INS or INSERT	{INSERT} or {INS}
LEFT ARROW	{LEFT}
NUM LOCK	{NUMLOCK}
PAGE DOWN	{PGDN}
PAGE UP	{PGUP}
PRINT SCREEN	{PRTSC} (reserved for future use)
RIGHT ARROW	{RIGHT}
SCROLL LOCK	{SCROLLLOCK}
TAB	{TAB}
UP ARROW	{UP}
F1	{F1}
F2	{F2}
F3	{F3}

**Table 1**

<b>Key</b>	<b>Code</b>
F4	{F4}
F5	{F5}
F6	{F6}
F7	{F7}
F8	{F8}
F9	{F9}
F10	{F10}
F11	{F11}
F12	{F12}
F13	{F13}
F14	{F14}
F15	{F15}
F16	{F16}
Keypad add	{ADD}
Keypad subtract	{SUBTRACT}
Keypad multiply	{MULTIPLY}
Keypad divide	{DIVIDE}

To specify keys combined with any combination of the SHIFT, CTRL, and ALT keys, precede the key code with one or more of the following codes.

**Table 2**

<b>Key</b>	<b>Code</b>
SHIFT	+
CTRL	^
ALT	%

To specify that any combination of SHIFT, CTRL, and ALT should be held down while several other keys are pressed, enclose the code for those keys in parentheses. For example, to specify to hold down SHIFT while E and C are pressed, use "+(EC)". To specify to hold down SHIFT while E is pressed, followed by C without SHIFT, use "+EC".

To specify repeating keys, use the form {key number}. You must put a space between key and number. For example, {LEFT 42} means press the LEFT ARROW key 42 times; {h 10} means press H 10 times.

----

Ett relaterat kommando är [A värde] som skriver ut tecknet med motsvarande ascii-värde, t.ex [A 9] betyder samma sak som {TAB} eller |.

## Avancerad automatisering: Introduktion

I Insert This V4.0 har det tillkommit ett flertal saker med vilka man kan skapa avancerad automatisering:

- I. Inspelning och uppspelning av musklick. Musklicken sparas som kommandon i Insert-This-textfönstret.
- II. ”Batch open and insert”: Man väljer ett antal filer i en fildialogruta (t.ex alla filer i en mapp). Dessa filer kommer sedan i tur och ordning att öppnas med sin standardapplikation och vald text i Insert-This-fönstret kommer att utföras för var och en av filerna.
- III. ”Batch from list”: Detta är en metod som gör att man utifrån ett register (en lista i en textfil) kan upprepa en procedur. Här väljer man en textfil som innehåller tabavgränsad data. (En sådan kan man t.ex exportera från Excel). Den valda texten i Insert-This-fönstret kommer då att utföras för varje rad i listan och innehållet från listraden får man in genom kommandot [C kolumnnummer]. (T.ex [c 1] för texten i kolumn 1).
- IV. Man kan köra en allmän loop där en räknare räknas upp. (Knappen ”For [i] to”).
- V. Villkorsstyrning. Beroende på en mängd olika villkor, bland annat beroende på innehållet på klippbordet, så kan olika sekvenser köras genom kommandona [if], [else] och [endif]. Kommandot [break] får nog också räknas in här.

Dessa olika möjligheter kommer nu att beskrivas mer utförligt.

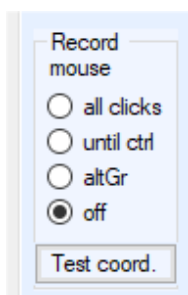
### Inspelning och uppspelning av musklick

Varning: Detta är en avancerad funktion som måste användas med försiktighet. När du spelar upp din inspelning så kommer den verkligen att klicka på skärmen på de inspelade positionerna helt oavsett av vad som visas på skärmen. Om du har framme ett annat program än det du spelade in på, så kommer den att klicka där istället, på det som visas på skärmen just nu. Om du har flyttat på fönstret du skall klicka i sedan du spelade in musklicken så kommer det inte heller att fungera för då kommer den ju att klicka där fönstret satt förut. Om ett oförutsett felmeddelande kommer upp från ett program så kommer det inte att fungera heller. Man måste kunna tänka logiskt och förutse allt som kommer att hända för att kunna använda denna funktion.

*För att avbryta en körning av en mängd musklick, håll inne vänster och höger musknapp samtidigt tills du får upp meddelandet ”Stopped by user”.*

Man kan spela in högerklick, vänsterklick samt dubbelklick men inte att man håller nere musknappen.

När man skall spela in musklick så ställer man först markören på den plats i Insert-This-textfönstret där man vill spela in ett eller flera musklick, sedan finns det tre olika sätt att spela in musklicken:



All Clicks: Alla (inspelningsbara) klick (enligt ovan) kommer att spelas in tills att du istället väljer ”Off”. Notera att även klicket för att trycka på ”Off” spelas in så det måste du sedan editera bort manuellt.

Until Ctrl: Alla klick spelas in tills dess att du trycker på Ctrl-knappen. Då växlar den till ”Off”.

AltGr: Bara de klick som sker när du håller nere AltGr-knappen kommer att spelas in. Notera att AltGr kan ha någon funktion i det program som du använder och i så fall rekommenderas inte detta alternativ.

Det som blir inspelat kommer att se ut ungefär såhär:

```
[L 1169 1060][L 1295 962][R 341 165][L 413 159]
```

R betyder höger musknapp (right). L betyder vänster knapp (left). Och siffrorna efter är koordinaterna för musklicket. T.ex [L 300 200] betyder ett klick med vänster musknapp vid koordinaten X=300 och Y=200 pixlar.

Om man då sedan markerar denna text och trycker på "Insert" så kommer musklicken att utföras.

Om man vill att någon text skall skrivas in mellan musklicken så får man skriva in det i Insert-This-fönstret. Det spelas alltså inte in automatiskt på samma sätt som musklicken. Så om vi säger att ".txt" skall skrivas in efter det första musklicket i exemplet ovan så blir det enligt följande:

```
[L 1169 1060].txt[L 1295 962][R 341 165][L 413 159]
```

Tänk på att även radbrytningar mellan muskommandon kommer att sättas in som text, så se till att alla muskommandon är på samma rad om du inte vill ha in oönskade radbrytningar. Det kan vara bra att ha på "Wrap" i editorn så att man ser all text.

*Obs: Om någon text är markerad när man börjar spela in musklick så kommer den INTE att skrivas över av musklicken. Om du ville det så radera texten först istället.*

## **Wait a moment**

Ibland så behövs en extra fördröjning efter att man klickat någonstans, t.ex för att ett program skall laddas in. Då kan man använda kommandot [W sekunder] t.ex [W 4] för att vänta i 4 sekunder. Antalet sekunder behöver inte vara ett heltal, så t.ex [W 0,5] kan användas för att vänta i en halv sekund. Observera att denna fördröjning påverkas av "Delay Adjustment"-reglaget som påverkar alla fördröjningar i Insert This. Om du verkligen vill vänta det antal sekunder som du specificerar så kan du istället använda WF, t.ex [WF 1] för att vänta i en sekund oavsett hur "Delay Adjustment"-reglaget är inställt. (WF står för Wait Force).

## **REM**

Om du vill skriva en kommentar i ditt program för att i efterhand förstå vad en viss sekvensbit gör, så använd kommandot [rem din kommentar]. Detta kommando hoppas över helt i exekveringen. Rem betyder "remark".

## **Musklick som utgår ifrån nuvarande position**

Kommandona L+ och R+ kan användas för att flytta musen utifrån nuvarande position och där utföra ett musklick. T.ex [L+ 0 16] kommer att flytta musen 16 pixlar nedåt och sedan klicka på vänster musknapp medan [R+ -8 0] kommer att flytta musen 8 pixlar åt vänster och sedan utföra högerklickning.

## **Testa koordinater**

Knappen "Test coord." tar de första muskoordinaterna i markerad text och sätter musen där utan att klicka. Meningen med detta är för att testa och för att se om ett kommando klickar på rätt ställe utan att den klickar där. Observera att denna funktion struntar i vilket kommando som används på koordinaterna, så koordinater till kommandona L+ och R+ (ovan) kommer att ses som normala (absoluta) koordinater istället.

## Batch open and insert

Med denna funktion så kan man utföra samma textändring, musklicksuppspelning (eller en kombination) på ett antal likadana filer, som alla kommer att öppnas med sin standardapplikation.

Detta är avancerat så läs denna instruktion noga och bedöm om du vågar dig på detta.

Tillvägagångssätt:

Gör först en kopia på alla filer som du tänker ändra, utifall något går fel i din programmering (och det gör det säkert när du håller på att testa). Gör en kopia på hela mappen.

Sedan får du kolla att det verkligen är det program du vill använda som är standardapplikation. Det gör du genom att dubbelklicka på en av filerna i utforskaren. Om inte rätt program öppnar filen så får du ändra standardapplikation för filtypen i Windows inställningar. (Hur man gör det tänker jag inte ta upp här. Sök på internet).

Skapa sedan en passande ändringssekvens i Insert This, som går att utföra precis när filen har öppnats. Om ditt syfte är att ändra i filen så måste sekvensen naturligtvis innehålla kommandon för att spara filen på slutet, kanske med ett nytt namn. Efter det måste du även ha med något som stänger filen igen, annars kommer du ha lika många öppna program som filer till slut.

Du kan testa din sekvens genom att markera sekvensen i Insert This, sedan dubbelklicka på en testfil i windows utforskare och sedan trycka på AltGr+Shift för att utföra sekvensen.

Testa sedan på bara några filer i en testmapp. T.ex 10 stycken filer. Markera sekvensen i Insert This och tryck sedan på "Batch open and instert", välj dina testfiler, observera och undersök om resultatet blev som du förväntade. Medan filerna öppnas en efter en och sekvensen körs för var och en av dem så kommer en grön stapel att gå längst uppe på skärmen för att visa hur långt det har gått. Om du märker att något går fel så kan du hålla inne både höger och vänster musknapp samtidigt. Då avbryts det hela. Håll inne knapparna hela tiden tills det avbryts.

Det sker automatiskt en paus på ett par sekunder när varje fil öppnas. Om denna paus inte räcker till för att filen skall hinna öppnas så kan du sätta ett wait-kommando som första kommando i din sekvens. T.ex [WF 5] (se tidigare beskrivning).

När du har genomfört ett lyckat test med flera filer så kan du köra sekvensen på alla filer som du tänkt. Men se först till att du inte har några andra program igång som kan störa det hela till exempel genom att plötsligt visa något meddelande på skärmen.

Här följer ett exempel på hur det kan se ut. Här är en sekvens som ändrar två datum i ett formulär i Word och sedan sparar filen med ett nytt namn genom en kombination av musklick och text:

```
[L 157 14][L 204 55][L 617 237]{TAB 16}2021-01-01|2021-12-31[L 29 48][L 62 249][L 566 194][w 1][L 1052 515]{left 5}-  
{( )ändrad( )}{enter}[w 1][L 1897 13]
```

Här har både Insert-This-kommandon och Microsoft-teckenkoder använts (se tidigare avsnitt). Insert-this-kommandon innesluts alltid av []-parenteser medan microsoft-teckenkoder innesluts i {}-parenteser. Om du vill skriva en normal parentes någonstans, så måste den skrivas { ( ) } eftersom vanliga parenteser har en speciell innebörd annars (se avsnittet om Microsofts teckenkoder).

Jag skall nu visa vilka delar av ovanstående exempel som gör vad genom att använda olika färger så kanske du förstår det hela lite mer:



[L 157 14][L 204 55][L 617 237]{TAB 16}2021-01-01|2021-12-31[L 29 48][L 62 249][L 566 194][w 1][L 1052 515]{left 5}-  
{(}ändrad{)}{enter}[w 1][L 1897 13]

Denna färg: Använder musen för att klicka på Visa och sedan Redigera dokument (i Word).

Klickar i filens första formulärsfält (är antagligen onödigt).

Tabbar sig fram till fältet som skall fyllas i och fyller där i ett datum, tabbar till nästa fält och fyller där i ett datum till.

Klickar sig fram till fildialogrutan för att spara i nuvarande mapp och väntar en sekund extra på detta.

Klickar i filnamnsrutan efter filnamnet. Går fem steg till vänster med markörpilarna för att gå förbi .docx extensionen och skriver där –(ändrad) och trycker sedan på enter för att spara.

Väntar sedan en extra sekund på detta.

Stänger fönstret med ett klick på krysset.

## Batch from list

Detta är en metod som gör att man utifrån ett register (en lista i en textfil) kan upprepa en procedur. Man markerar först den text i insert-this fönstret som skall utföras för varje rad i registerfilen. Sedan trycker man på knappen ”Batch from list” och väljer registerfilen i fråga.

Registerfilen skall vara en textfil som innehåller tab-avgränsad data. (En sådan kan man t.ex exportera från Excel).

Hur data kommer in från registerfilen är genom att man i instert-this använder kommandot [C kolumnnummer]. (T.ex [c 1] för texten i kolumn 1).

Här kommer ett förklarande exempel:

En registerfil ser ut enligt följande:

<i>Olof</i>	<i>Petterson</i>	<i>Ekeby</i>
<i>Ölme</i>	<i>Ylleman</i>	<i>Grabohusgårde</i>
<i>Erik</i>	<i>Svennhög</i>	<i>Grumsbro</i>

Markerad text i Instert-This-fönstret ser ut enligt följande:

```
Förnamn: [c 1]  
Efternamn: [c 2]  
Ort: [c 3]  
----{enter}
```

Om man sedan ”utför” den markerade texten i en texteditor, så får man följande resultat:

```
Förnamn: Olof  
Efternamn: Petterson  
Ort: Ekeby  
----  
Förnamn: Ölme  
Efternamn: Ylleman  
Ort: Grabohusgårde  
----  
Förnamn: Erik
```

Efternamn: Svennhög

Ort: Grumsbro

----

Meningen är att man genom att även använda musklick i sin procedur t.ex kan skapa ett antal dokument utifrån en mall. Proceduren i Insert-This-texten skulle då kunna ladda in en mall i t.ex word, fylla i data från listan, spara filen där ett av fälten i listan används som del i filnamnet, t.ex personnummer och sedan stänga filen. Sedan får man lika många filer som rader i listfilen.

Det skulle också kunna användas till att gå in i en databas, till exempel Pro Capita, och ändra något för alla personer i listan. Då skulle listan innehålla personnummer och man skulle använda musklicksprogrammering för att söka upp varje person och göra ändringen i databasen.

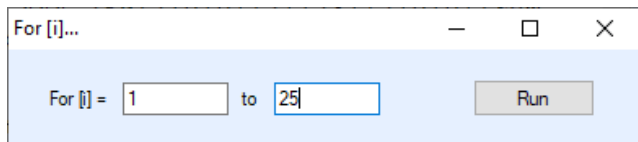
Pågående körning avbryts på samma sätt som ovanstående funktion, alltså genom att hålla nere båda musknapparna samtidigt tills det avbryts.

*Varning: Samma försiktighet måste tillämpas som beskrevs under "Batch open and insert"!*

## FOR-loop

Både "Batch open and insert" samt "Batch from list" i föregående avsnitt är ju olika sätt att upprepa en sekvens ett flertal gånger. Här kommer en tredje metod som är en lite mer generell loop.

Här markerar du först den sekvens i texten som skall köras. Sedan trycker du på knappen "For [i] to" och får då upp en ruta som ser ut såhär:



Om du låter den första rutan vara 1 så är den andra rutan antalet gånger loopen kommer att köras, i detta fall 25 gånger. För varje varv så räknas variabeln [i] upp med 1 och denna variabel går att använda både i text och i villkorsstyrning (se nästa avsnitt).

Om du t.ex skriver 20 i första rutan och 24 i andra rutan och kör sekvensen "Hej [i]{enter}" så kommer följande att skickas till tangentbordet:

Hej 20  
Hej 21  
Hej 22  
Hej 23  
Hej 24

Pågående körning avbryts på samma sätt som övriga loop-funktioner, alltså genom att hålla nere båda musknapparna samtidigt tills det avbryts.

## Villkorsstyrning

Kommandona för villkorsstyrning ser ut enligt följande:

```
[if uttryck] ...sekvens som körs om uttrycket stämmer... [else] ...sekvens som körs om uttrycket inte stämmer... [endif]
```

Utan else:

```
[if uttryck] ...sekvens som körs om uttrycket stämmer... [endif]
```

”Uttrycket” är uppbyggt enligt:

**Parameter\_A jämförelsemetod parameter\_B**

Möjliga parametrar är:

- Clip** Det är textinnehållet på klippbordet som jämförs.
- Filename** Det är aktuellt filnamn som jämförs. (Enbart när Batch open and insert används)
- C nr** Data från kolumn nr nr. (Enbart när Batch from list används)
- i** Aktuell siffra om du kör en FOR [i]-loop.
- Annat\_ord** Vilken text eller siffra som helst.
- ”Fler ord”** Om mellanslag finns i texten du vill jämföra så inneslut texten med ”-tecken.

*Notera: Om man vill ta bort betydelsen av specialparametrarna, så inneslut dem med ”-tecken. T.ex ”filename” betyder texten filename och inte aktuellt filnamn.*

*Notera också: Microsofts teckenkoder kan inte användas inne i parametrar.*

**Jämförelsemetoder:**

- A = B** Sant om A är lika med B
- A <> B** Sant om A inte är lika med B
- A contains B** Sant om A innehåller B, t.ex om B är ”hej” och A är ”Hej på dig” så innehåller A B.
- A > B** Sant om A är större än B, t.ex om A är 5 och B är 2.
- A < B** Sant om A är mindre än B
- A >= B** Sant om A är större eller lika med B
- A <= B** Sant om A är lika eller mindre än B

Det måste finnas mellanslag mellan parametrarna och jämförelsemetoden.

OBS: Villkorssatser kan inte vara nästlade. Med andra ord så kan man inte ha en ny if-sats inne i en annan if-else-endif-konstruktion.

Kommandot **[break kommentar]** används för att avbryta exekveringen samt pågående loop eller batch och används främst ihop med villkor, t.ex.

```
[if clip <> 201231][break Felaktigt datum][endif]
```

Några exempel:

*Detta skriver olika text beroende på om klippbordet innehåller ordet "sommar" eller inte:*

```
[if clip contains sommar] Ordet sommar finns i texten på klippbordet [else] Sommar finns inte på klippbordet [endif]
```

*För att lägga något på klippbordet, använd teckenkoden ^C (se avsnittet teckenkoder). En sekvens som först lägger något på klippbordet och sedan jämför det skulle kunna se ut såhär:*

```
[L 617 237]^C[if clip = 2020-07-01] ...Gör något... [endif]
```

*Jämförelse av tal i kolumn 2 i sin datalista (när man använder Batch from list)*

```
[if c 2 > 200] Data i kolumn 2 är större än 200 [endif]
```

*Olika sekvenser beroende på filtyp vid användning av Batch open and insert:*

```
[if filename contains .docx] ...sekvens för word... [endif][if filename contains .txt] ...sekvens för Notepad... [endif][if filename contains .xlsx] ...sekvens för excel... [endif]
```

*Olika sekvenser beroende på variabeln i:s värde vid användning av FOR-loop:*

```
[if i > 99][break För stort tal][endif][if i < 10] ...sekvens för tal under 10... [else] ...sekvens för tal 10 till 99... [endif]
```

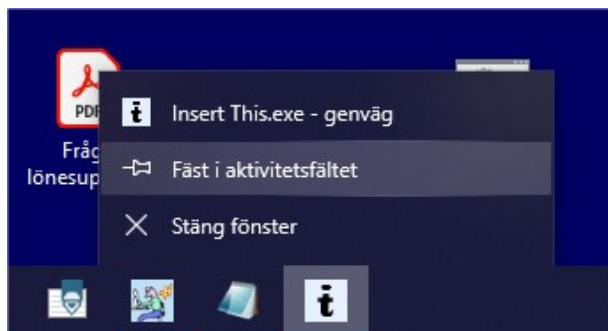
*Följande avancerade exempel är nog lite svårbegripligt men det finns med för att visa möjligheterna. Utifrån en lista med personer i Excel så går den över till Pro Capita och avslutar en "aktivitet" i en plan samt startar en ny. Sedan går den tillbaka till Excel och sätter ett kryss för personen i listan. Sekvensen körs genom att ställa markören på första personnumret i Excel och sedan startar man sekvensen med "For [i] to" där man matar in första excel-radnumret samt sista radnumret. Klippbordet används både till att få personnumret från Excel till Pro Capita samt för att kolla att man hamnat på rätt "aktivitet".*

Förlängning i Pro Capita

```
^c[L 857 1061][L 892 973][L 191 88]^A^V{enter}[w 1][L 668 434][L 928 670][L 186 203][w 2][L 347 133][w 4][L 438 288][w .3]^c[w .3][if clip <> 201231][break Fel slutdatum eller aktivitet][endif][L 974 288]201231[L 976 309][L 872 356][L 43 55][w 1][L 16 58][w 1][L 609 165][L 503 199][L 907 165][L 862 199][L 885 197][L 838 224][L 444 289]210101|211231|210101[rem plast och handläggare][L 514 195][L 493 339][L 516 223][L 616 569][L 499 498][rem spara][L 40 57][w 5][rem kryssa][L 469 1064]{tab 7}x{enter}
```

## Förslag på installation

1. Ladda ner programmet från <http://www.boray.se/software/insertthis/>
2. Spara det på skrivbordet.
3. Starta programmet genom att dubbelklicka på dess ikon på skrivbordet.
4. Gå ner till aktivitetsfältet och **högerklicka** på Insert-this-ikonen. Välj ”Fäst i aktivitetsfältet”



5. Klart