

```
1  #!/bin/bash
2
3  trap control_c SIGINT
4  control_c()
5  {
6      echo -e "\e[0m"
7      echo -e "Exited on Signal 2. Job" "\e[1;31m" "ABORTED." "\e[0m"
8      sleep 1
9      clear
10     stty echo
11
12     if [ "$jobno" ]
13     then
14         mysql -N pos -e "delete from posdata where job_no = '$jobno';"
15         echo "Work Order: $jobno Deleted."
16     fi
17     exit 99
18 }
19
20
21 jobno=""
22 # tmpdir="/home/pos/tmp/"
23 tmpdir="/home/pos/tmp/"
24 etcdir="/home/pos/etc/"
25 logdir="/home/pos/log/"
26 outdir="/home/pos/archive/"
27
28
29 # stty erase
30
31 clear
32
33
34 # Get Info From Keyboard
35 echo "Enter Existing Customer Name,"
36 echo "Or Blank For New:"
37 read name
38
39 if [ -z "$name" ]
40 then
41     name="BCC-New-Customer"
42 fi
43
44
45 dbname=`mysql -N pos -e "select customer_name from posdata where customer_name like
46     '%$name%' limit 1; "`
47
48 if [ -n "$dbname" ]
49 then
50     echo $dbname " Y/N?"
51
52     read ans
53
54     case "$ans" in
55         "Y" | "y")
56             name=$dbname
57             addr=`mysql -N pos -e "select customer_address from posdata where customer
58             _name like '%$name%' limit 1; "`
59             city=`mysql -N pos -e "select city from posdata where customer_name like '
60             %$name%' limit 1; "`
61             zip=`mysql -N pos -e "select zip from posdata where customer_name like '%$
62             name%' limit 1; "`
63             phone=`mysql -N pos -e "select customer_phone from posdata where customer_
64             name like '%$name%' limit 1; "`
65             ;;
66         *)
67             echo "Enter Customer Name"
```

```

65         read name
66         echo "Enter Address"
67         read addr
68         echo "Enter City"
69         read city
70         echo "Enter Zip"
71         read zip
72         echo "Enter Fone"
73         read phone
74         ;;
75     esac
76
77     else
78         if [ "$name" = "BCC-New-Customer" ]
79         then
80             modifier=""
81         else
82             modifier="None Found. "
83         fi
84
85         echo ${modifier}"Enter New Customer Name"
86         read name
87         echo "Enter Address"
88         read addr
89         echo "Enter City"
90         read city
91         echo "Enter Zip"
92         read zip
93         echo "Enter Fone"
94         read phone
95
96     fi
97
98
99     csz=$city-FL-$zip
100
101     lastjob=`mysql -N pos -e "select job_no from posdata order by job_no desc limit 1;"`
102
103     jobno=`expr $lastjob + 1`
104
105     if [ "$name" = "BCC Test Job" ]
106     then
107         jobno=00000
108         mysql -N pos -e "delete from posdata where job_no = '$jobno';"
109     fi
110
111
112     # Zero Pad Job Number
113     num=0000000${jobno}
114     len=`echo $num | wc -c`
115     digs=`expr $len - 5`
116     jobno=`echo $num | cut -c ${digs}-`
117
118
119     echo "Jobno is "$jobno
120
121     # Get Job Number from MySQL Database
122
123     # Start with new template
124
125     cat ${etcdir}Work-Order-Auth.template.ps > Work-Order-Auth
126
127
128     # Set Date, working file, invoice number
129     date=`date "+%b %d, %Y"`
130     woapdf="./Work-Order-Auth"
131     invoiceno="-----"
132

```

```

133 mysql pos -e "insert into posdata (job_no, invoice_no, customer_name, customer_addr
    ess, city, state, zip, customer_phone) values('$jobno', '$invoiceno', '$name', '$ad
    dr', '$city', 'FL', '$zip', '$phone');"
134
135 # Get and set length of various input data for PostScript
136 namecount=`echo -n $name | wc -c`
137 namesize=`expr $namecount \* 85`
138
139 addrcount=`echo -n $addr | wc -c`
140 addrsize=`expr $addrcount \* 85`
141
142 cszcount=`echo -n $csz | wc -c`
143 cszsize=`expr $cszcount \* 85`
144
145 phonecount=`echo -n $phone | wc -c`
146 phonesize=`expr $phonecount \* 85`
147
148 jobnocount=`echo -n $jobno | wc -c`
149 jobnosize=`expr $jobnocount \* 275`
150
151 invoicenocount=`echo -n $invoiceno | wc -c`
152 invoicenosize=`expr $invoicenocount \* 235`
153
154 datecount=`echo -n $date | wc -c`
155 datesize=`expr $datecount \* 85`
156
157 # linecount=`echo -n $line | wc -c`
158 # linesize=`expr $linecount \* 75`
159
160
161 cusname="($name ) $namesize"
162 cusaddr="($addr ) $addrsize"
163 cuscsz="($csz ) $cszsize"
164 cusphone="($phone ) $phonesize"
165 # cusjobno="(Job No: $jobno) $jobnosize"
166 cusinvoiceno="(Invoice No: $invoiceno) $invoicenosize"
167 cusdate="($date) $datesize"
168
169
170 # Enter customer information, etc into work order
171 sed -i -e "s/(Customer-Name) 1200/$cusname/" $woapdf
172 sed -i -e "s/(Customer-Street-Address) 1725/$cusaddr/" $woapdf
173 sed -i -e "s/(City-State-Zip) 1725/$cuscsz/" $woapdf
174 sed -i -e "s/(Phone-Number) 900/$cusphone/" $woapdf
175 sed -i -e "s/(Date) 1020/$cusdate/" $woapdf
176 sed -i -e "s/Job-No/$jobno/" $woapdf
177 sed -i -e "s/Invoice: _____/ /" $woapdf
178
179
180 # Get Description and process it
181 rm -f descr.$jobno
182 rm -f ${tmpdir}descr.$jobno.tmp
183
184
185 ## Old Method
186 # echo "Enter Description"
187 #
188 # while read line
189 # do
190 #     # echo $line >> ${tmpdir}descr.$jobno.tmp
191 #     echo $line | tr -d "'" | tr -s "&" "+" | tr -s ")" ":" | tr -s "(" ":" >> $
    {tmpdir}descr.$jobno.tmp
192 # done
193
194
195 ## New Method
196 # Get Info From Barcode Scanner
197 echo -e "\e[0m"
198 echo -e "Scan Barcodes. Scan" "\e[1;31m" "DONE" "\e[0m" "When Done"

```

```
199
200     stty -echo
201 while read line
202 do
203
204     if [ "$line" = "DONE" ]
205     then
206         echo "Done"
207         break
208     fi
209
210     if [ "$line" = "APPLE" ]
211     then
212         echo "APPLE"
213     fi
214
215     if [ "$line" = "OTHER" ]
216     then
217         # echo "Enter Type, then hit Enter to Continue... "
218         # stty echo
219         # read line
220         # echo $line | tr -d '"' | tr -s "&" "+" | tr -s ")" ":" | tr -s "(
221 " ":" >> ${tmpdir}descr.${jobno}.tmp
222         # echo -e "Next Barcode, or " "\e[1;31m" "DONE" "\e[0m" " to Con
tinue"
223
224         # stty -echo
225         # continue
226         echo "OTHER"
227         line="OTHER"
228     fi
229
230     if [ "$line" = "SONY" ]
231     then
232         echo "SONY"
233         line="SONY"
234     fi
235
236     if [ "$line" = "GTWY" ]
237     then
238         echo "GATEWAY"
239         line="GATEWAY"
240     fi
241
242     if [ "$line" = "DSKTP" ]
243     then
244         echo "Desktop"
245         line="Desktop"
246     fi
247
248     if [ "$line" = "LPTP" ]
249     then
250         echo "Laptop"
251         line="Laptop"
252     fi
253
254     if [ "$line" = "DELL" ]
255     then
256         echo "DELL"
257     fi
258
259     if [ "$line" = "IBM" ]
260     then
261         echo "IBM"
262     fi
263
264     if [ "$line" = "CPQ" ]
265     then
266         echo "COMPAQ"
267         line="COMPAQ"
```

```

266         fi
267
268         if [ "$line" = "HP" ]
269         then
270             echo "Hewlett-Packard"
271             line="Hewlett-Packard"
272         fi
273
274         if [ "$line" = "TOS" ]
275         then
276             echo "Toshiba"
277             line="Toshiba"
278         fi
279
280         if [ "$line" = "DGNCL" ]
281         then
282             echo "Diagnose and Call"
283             line="Diagnose and Call"
284         fi
285
286         if [ "$line" = "PWADP" ]
287         then
288             echo "Has Power Adapter"
289             line="Has Power Adapter"
290         fi
291
292         if [ "$line" = "GOAHD" ]
293         then
294             echo "GO Ahead And Fix"
295             line="GO Ahead And Fix"
296         fi
297
298         if [ "$line" = "ACER" ]
299         then
300             echo "ACER"
301         fi
302
303         if [ "$line" = "NPADP" ]
304         then
305             echo "NO Power Adapter"
306             line="NO Power Adapter"
307         fi
308
309         if [ "$line" = "PRTNO" ]
310         then
311             echo -n "Part Number: "
312             stty echo
313             line="Part Number: "
314             echo -n $line | tr -d "'" | tr -s "&" "+" | tr -s ")" ":" | tr -s "
315 (" ":" >> ${tmpdir}descr.${jobno}.tmp
316                 continue
317             fi
318
319         if [ "$line" = "SERNO" ]
320         then
321             echo -n "Serial Number:"
322             stty echo
323             line="Serial Number: "
324             echo -n $line | tr -d "'" | tr -s "&" "+" | tr -s ")" ":" | tr -s "
325 (" ":" >> ${tmpdir}descr.${jobno}.tmp
326                 continue
327             fi
328
329         if [ "$line" = "SVTAG" ]
330         then
331             echo -n "Service Tag: "
332             stty echo
333             line="Service Tag: "
334             echo -n $line | tr -d "'" | tr -s "&" "+" | tr -s ")" ":" | tr -s "

```

```

333     (" ":" >> ${tmpdir}descr.${jobno}.tmp
334         continue
335     fi
336     if [ "$line" = "SLOW" ]
337     then
338         echo "SLOW"
339         line="SLOW"
340     fi
341
342     if [ "$line" = "OPTMZ" ]
343     then
344         echo "Check For Optimization."
345         line="Check For Optimization."
346     fi
347
348     if [ "$line" = "VIRUS" ]
349     then
350         echo "Virus Check."
351         line="Virus Check."
352     fi
353
354     if [ "$line" = "BSOD" ]
355     then
356         echo "Blue Screen"
357         line="Blue Screen of Death"
358     fi
359
360     echo $line | tr -d "'" | tr -s "&" "+" | tr -s ")" ":" | tr -s "(" ":" >> $
361     {tmpdir}descr.${jobno}.tmp
362     stty -echo
363 done
364 stty echo
365
366 echo -e "Enter Additional Info If Any... Or Scan" "\e[1;31m" "DONE" "\e[0m" "If D
367 one."
368 while read line
369 do
370     if [ "$line" = "DONE" ]
371     then
372         echo "Done"
373         break
374     fi
375     echo $line | tr -d "'" | tr -s "&" "+" | tr -s ")" ":" | tr -s "(" ":" >> $
376     {tmpdir}descr.${jobno}.tmp
377 done
378
379 elvfmt -s -w 64 ${tmpdir}descr.${jobno}.tmp | tr "/" "-" > descr.${jobno}
380
381
382 # paginate description and enter it ino
383 # body of form
384 linecount=`cat descr.${jobno} | wc -l`
385 pagecount=`expr $linecount / 35`
386 pagecount=`expr $pagecount + 1`
387
388 startline=1
389
390 for page in `seq 1 $pagecount`
391 do
392     # setup pages
393     prnpage=$page
394     num=0000000${page}
395     len=`echo $num | wc -c`
396     digs=`expr $len - 4`
397     page=`echo $num | cut -c ${digs}-`

```

```

398
399     echo "Page $page"
400
401     cp $woapdf $woapdf.$jobno.$page.ps
402
403     lastline=`expr $startline + 34`
404
405     sed -n "${startline},${lastline}p" descr.$jobno > descr.$jobno.$page
406
407
408     descline=1
409
410     sed -i -e "211s/(                ) 7650 X/( Page: $prnpage of $pagecount) 100
0 X/" $woapdf.$jobno.$page.ps
411
412
413
414
415     # In each page, place page from description
416     for lineno in `cat ${etcdir}lines.template.body | awk -F\: '{ print $1 }' |
head -n 35`
417
418     do
419
420         echo -n $lineno
421         linelen=`sed -n "${descline}p" descr.$jobno.$page | wc -c`
422         psdata=`sed -n "${descline}p" descr.$jobno.$page`
423
424         pslength=`expr $linelen \* 100`
425         sed -i -e "${lineno}s/(                ) 7650 X/( $psdata ) $pslength
X/" $woapdf.$jobno.$page.ps
426
427
428         echo "    "$descline"    "$pslength
429
430
431         descline=`expr $descline + 1`
432
433     done
434
435
436
437     startline=`expr $lastline + 1`
438
439     ps2pdf $woapdf.$jobno.$page.ps > ${logdir}wo.$jobno.$page.out 2>&1
440     # mv $jobno.$page.out $logdir
441     mv $woapdf.$jobno.$page.ps $logdir
442
443     mv $woapdf.$jobno.$page.pdf $outdir
444
445
446     mv descr.$jobno.$page $logdir
447
448     description=`cat descr.$jobno | tr -d "&" | tr -d "' '`
449
450
451 done
452 # End of Pagination Routine
453
454 currdesc=`mysql pos -N -e "select workorder_text from posdata where job_no = '$jobno' and workorder_text is not null;"`
455 mysql pos -e "update posdata set workorder_text = '$currdesc' \n '$description' where job_no = '$jobno';"
456
457 ### Need to catenate all pages in to one
458 ### As with invoice.sh
459 mv descr.$jobno $logdir
460 mv $woapdf ${logdir}$woapdf.$jobno
461

```

```
462 for f in ${outdir}$woapdf.$jobno.$page.pdf
463 do
464 # xpdf $f
465 xpdf -geometry 800x700+200+0 $f 2> /dev/null
466 done
467
468
469 workorder_datetime=`date +%Y-%m-%d" "%H:%M:%S`
470
471 mysql pos -e "update posdata set workorder_datetime = '$workorder_datetime' where
    job_no = '$jobno'"
472
473
474
```