

```
1  #!/bin/bash
2
3  trap control_c SIGINT
4
5  #Exit Function
6  # run if user hits control-c
7  control_c()
8  {
9      # stty sane
10     echo -e "\e[0m"
11     echo -e "Exited on Signal 2. Transaction" "\e[1;31m" "ABORTED." "\e[0m"
12     sleep 1
13     clear
14
15     # stty erase
16     # reset
17     exit 98
18 }
19
20
21 servicetotal=0
22 taxabletotal=0
23 cost=0
24 coststring=0
25 sum=0
26 sumstring=0
27 itemstring=""
28 item=""
29 service=""
30
31 # tmpdir="/home/pos/tmp/"
32 tmpdir="/home/pos/tmp/"
33 etcdir="/home/pos/etc/"
34 logdir="/home/pos/log/"
35 outdir="/home/pos/archive/"
36
37
38 # stty erase
39 clear
40
41 ## Get Job Number From Keyboard
42 echo "Enter Job No: "
43 read jobno
44
45 ## Zero Pad Job Number
46 num=0000000${jobno}
47 len=`echo $num | wc -c`
48 digs=`expr $len - 5`
49 jobno=`echo $num | cut -c ${digs}-`
50
51 echo "Jobno is "$jobno
52
53
54 # Start with new template
55 cat ${etcdir}Invoice.template.ps > Invoice
56
57
58 ## Set Date, working file, invoice number
59 date=`date "+%b %d, %Y"`
60 woapdf="./Invoice"
61 woaemail="./Invoice.email"
62 woa_all="./Invoice.all"
63 woa_form="./Invoice.print"
64
65 ## de-null data field
66 mysql pos -e "update posdata set invoice_text = '--' where job_no like '$jobno' ;"
67
68
69 ## Begin service and taxable handling
```

```

70
71 ##--Services Rendered
72 while :
73 do
74
75     echo -e "Enter\e[1;33m Service(s)\e[0m Rendered Or Leave\e[1m Blank\e[0m To
Continue: "
76     read service
77     if [ ! "${service}" ]
78     then
79         # servicetotal=0
80         break
81     fi
82
83     echo -n "Enter Cost: $"
84     read servicecost
85     echo "${servicecost}" | sed -e 's/\./\./' | grep -q [^[:digit:]]
86     if [ $? = 0 ]
87     then
88         echo -e -n "ERROR:\e[1;31m Non Numeric Value Given.\e[0m Re-"
89         servicecost=""
90     fi
91
92     while [ ! "${servicecost}" ]
93     do
94         echo -n "Enter Cost: $"
95         read servicecost
96         if [ ! "${servicecost}" ]
97         then
98             continue
99         else
100             echo "${servicecost}" | sed -e 's/\./\./' | g
rep -q [^[:digit:]]
101             if [ $? = 0 ]
102             then
103                 echo -e -n "ERROR:\e[1;31m Non Numeric Valu
e Given.\e[0m Re-"
104                 servicecost=""
105                 continue
106             fi
107         fi
108     done
109
110     # echo ${service} " - " ${cost}
111     servicetotal=`echo "$servicetotal $servicecost" | awk '{ sum = $1 + $2; pri
ntf "%.2f",sum }'`
112     servicedatastring=${servicedatastring}${service}%
113     servicecoststring=${servicecoststring}${servicecost}+
114
115     echo
116     mysql pos -e "update posdata set services_rendered = '${servicedatastring}' where j
ob_no like '$jobno' ;"
117     mysql pos -e "update posdata set service_charges = '${servicecoststring}' where job
_no like '$jobno' ;"
118 done
119 ##--End Services Rendered
120
121
122
123
124 ##-- Taxable Items
125 while :
126 do
127
128     echo -e "Enter\e[1;34m Taxable\e[0m Item(s) Or Leave Blank To Continue: "
129     read taxableitem
130     if [ ! "${taxableitem}" ]
131     then
132         # taxabletotal=0

```

```

133             taxstr="Taxable Items: None."
134             break
135         fi
136     echo -n "Enter Cost: $"
137     read taxablecost
138
139     echo "${taxablecost}" | sed -e 's/\./\./' | grep -q [^[:digit:]]
140     if [ $? = 0 ]
141     then
142         echo -e -n "ERROR:\e[1;31m Non Numeric Value Given.\e[0m Re-"
143         taxablecost=""
144     fi
145
146     while [ ! "${taxablecost}" ]
147     do
148         echo -e -n "Enter Cost: $"
149         read taxablecost
150         if [ ! "${taxablecost}" ]
151         then
152             continue
153         else
154             echo "${taxablecost}" | sed -e 's/\./\./' | g
155             if [ $? = 0 ]
156             then
157                 echo -e -n "ERROR:\e[1;31m Non Numeric Valu
158                 e Given.\e[0m Re-"
159                 taxablecost=""
160                 continue
161             fi
162             if [ ! "${taxablecost}" ]
163             then
164                 taxstr="Taxable Items: None."
165                 continue
166             fi
167         done
168
169         taxabletotal=`echo "$taxabletotal $taxablecost" | awk '{ sum = $1 + $2; pri
170         ntf "%.2f",sum }'`
171         # echo ${item} " - " ${cost}
172         taxableitemstring=${taxableitemstring}${taxableitem}%
173         taxablecoststring=${taxablecoststring}${taxablecost}+
174         # echo ${itemstring} " ; " ${coststring}
175         taxstr="Taxable Items: "
176
177         # echo "Enter Taxable Item: "
178         # continue
179
180     echo
181     mysql pos -e "update posdata set parts_used = '${taxableitemstring}' where job_no l
182     ike '$jobno' ;"
183     mysql pos -e "update posdata set parts_costs = '${taxablecoststring}' where job_no
184     like '$jobno' ;"
185     done
186     ##--End Taxable Items
187
188     ## Print Out Subtotals
189     echo "servicetotal ="$servicetotal
190     echo "taxabletotal ="$taxabletotal
191
192     ## Exit If Neither Entered
193     if [ `echo $servicetotal | tr -d "."` -eq 0 ] && [ `echo $taxabletotal | tr -d "."`
194     -eq 0 ]
195     then
196         echo -e "No Items and No Service.\e[1;31m Exiting\e[0m"
197         exit 1
198     fi

```

```

196
197 ### End service and taxable handling
198
199
200 ## Generate New Invoice Number and
201 ## Enter it in to database
202 # select existing data from mysql
203 # invoiceno="-----"
204 # invoiceno="00020"
205 lastinvoiceno=`mysql -N pos -e "select invoice_no from posdata order by 1 desc lim
it 1;"`
206 invoiceno=`expr $lastinvoiceno + 1`
207 echo "Invoice Number is: "$invoiceno
208 echo "Including Info From Work Order # "$jobno
209
210 name=`mysql -N pos -e "select customer_name from posdata where job_no like '$jobno'
;"`
211 addr=`mysql -N pos -e "select customer_address from posdata where job_no like '$job
no';"`
212 csz=`mysql -N pos -e "select city, state, zip from posdata where job_no like '$jobn
o';"`
213 phone=`mysql -N pos -e "select customer_phone from posdata where job_no like '$jobn
o';"`
214 echo $phone > /tmp/foo
215
216 mysql pos -e "update posdata set invoice_no = '$invoiceno' where job_no like '$jobn
o';"
217
218
219 ## Get and set lenght of various input data for PostScript
220 namecount=`echo -n $name | wc -c`
221 namesize=`expr $namecount \* 85`
222
223 addrcount=`echo -n $addr | wc -c`
224 addrsize=`expr $addrcount \* 85`
225
226 cszcount=`echo -n $csz | wc -c`
227 cszsize=`expr $cszcount \* 85`
228
229 phonecount=`echo -n $phone | wc -c`
230 phonesize=`expr $phonecount \* 85`
231
232 jobnocount=`echo -n $jobno | wc -c`
233 jobnosize=`expr $jobnocount \* 275`
234
235 invoicenocount=`echo -n $invoiceno | wc -c`
236 invoicenosize=`expr $invoicenocount \* 235`
237
238 datecount=`echo -n $date | wc -c`
239 datesize=`expr $datecount \* 85`
240
241 # linecount=`echo -n $line | wc -c`
242 # linesize=`expr $linecount \* 75`
243
244
245 cusname="( $name ) $namesize"
246 cusaddr="( $addr ) $addrsize"
247 cuscsz="( $csz ) $cszsize"
248 cusphone="( $phone ) $phonesize"
249 # cusjobno="(Job No: $jobno) $jobnosize"
250 cusinvoiceno="(Invoice No: $invoiceno) $invoicenosize"
251 cusdate="( $date ) $datesize"
252
253 echo "Name is: ""$name"
254 echo "Name "
255 # echo "$cusaddr" $woapdf
256 # echo "$cuscsz" $woapdf
257 # echo "$cusphone" $woapdf
258 # echo "$cusdate" $woapdf

```

```

259 # echo "$jobno" $woapdf
260 # echo "$invoiceno" $woapdf
261
262 # Enter customer information, etc into work order
263 sed -i -e "s/(Customer-Name) 1200/$cusname/" $woapdf
264 sed -i -e "s/(Customer-Street-Address) 1725/$cusaddr/" $woapdf
265 sed -i -e "s/(City-State-Zip) 1725/$cuscsz/" $woapdf
266 sed -i -e "s/(Phone-Number) 900/$cusphone/" $woapdf
267 sed -i -e "s/(Date) 1020/$cusdate/" $woapdf
268 sed -i -e "s/Job-No/$jobno/" $woapdf
269 sed -i -e "s/Invoice#:      /Invoice#: 000$invoiceno/" $woapdf
270
271
272
273 rm -f invoice.$jobno
274 # rm -f ${tmpdir}invoice.$jobno.tmp
275 echo -n > $woa_all.$jobno.ps
276 echo -n > $woa_form.$jobno.ps
277
278
279 # Get Description and process it
280 # mysql -N pos -e "select workorder_text from posdata where job_no like '$jobno';"
> ${tmpdir}invoice.$jobno.tmp
281 mysql -N pos -e "select workorder_text from posdata where job_no like '$jobno';" |
sed -e "s/\\n/\\n/g" > ${tmpdir}invoice.$jobno.tmp
282 # mysql -N pos -e "select tech_notes from posdata where job_no like '$jobno';" | s
ed -e "s/NULL/\\n/g" >> ${tmpdir}invoice.$jobno.tmp
283
284
285
286 count=`echo ${servicedatastring} | tr "%" "\n" | wc -l`
287 count=`expr $count - 1`
288
289 # echo "count= "$count
290
291 intext=`mysql -N pos -e "select invoice_text from posdata where job_no like '$jobno
';"`
292 mysql pos -e "update posdata set invoice_text = '$intext \n Services: ' where job_n
o like '$jobno' ;"
293
294 for field in `seq 1 $count`
295 do
296     cst=`echo ${servicecoststring} | awk -F\+ "{ print \\${field} }"`
297     svc=`echo ${servicedatastring} | awk -F\% "{ print \\${field} }"`
298     echo ${field}: $svc" - "$cst
299     intext=`mysql -N pos -e "select invoice_text from posdata where job_no like
'$jobno' ;"`
300     mysql pos -e "update posdata set invoice_text = '$intext \n $field: $svc -
\\${cst} ' where job_no like '$jobno' ;"
301
302 done
303
304
305 count=`echo ${taxableitemstring} | tr "%" "\n" | wc -l`
306 count=`expr $count - 1`
307
308 intext=`mysql -N pos -e "select invoice_text from posdata where job_no like '$jobno
';"`
309 mysql pos -e "update posdata set invoice_text = '$intext \n $taxstr ' where job_no
like '$jobno' ;"
310
311 for field in `seq 1 $count`
312 do
313     prc=`echo ${taxablecoststring} | awk -F\+ "{ print \\${field} }"`
314     itm=`echo ${taxableitemstring} | awk -F\% "{ print \\${field} }"`
315     echo ${field}: $itm" - "$prc
316     intext=`mysql -N pos -e "select invoice_text from posdata where job_no like
'$jobno' ;"`
317     mysql pos -e "update posdata set invoice_text = '$intext \n $field: $itm -

```

```

    \${prc} ' where job_no like '$jobno' ;"
318
319 done
320
321
322
323 ## --- Start with routine here to create itemized invoice text
324 # mysql pos -e "update posdata set invoice_text = '${servicedatastring}:${serviceco
ststring}--${taxableitemstring}:${taxablecoststring}' where job_no like '$jobno' ;"
325
326
327
328 mysql -r -N pos -e "select invoice_text from posdata where job_no like '$jobno';" |
    sed -e "s/NULL/\n/g" | tr -d '"' | tr -s "&" "+" | tr -s ")" ":" | tr -s "(" ":" >
    > ${tmpdir}invoice.$jobno.tmp
329 elvfmt -s -w 64 ${tmpdir}invoice.$jobno.tmp | tr "/" "-" > invoice.$jobno
330
331
332 ##### Begin Payment Handling
333 ##Determine Payment Type
334 # echo -e "Enter Payment Type:\n""\e[1;33m""[C]" "\e[0m" "Credit" " --""\e[1;34m" "
[K]" "\e[0m" "Cash/Check"
335 echo -e "Enter Payment Type:\n""\e[1;33m""[C]" "\e[0m" "Credit" " --""\e[1;34m" " [
K]" "\e[0m" "Cash/Check" " --""\e[1;31m" " [I]" "\e[0m" "Invoice"" "
336 paytype=""
337
338 while :
339 do
340 echo -n -e "\e[0;30m"
341 stty raw
342 paytype=`dd bs=1 count=1 2> /dev/null`
343 stty -raw
344 echo -n -e "\e[0m"
345
346 case "$paytype" in
347     "k" | "K")
348         if [ "$keyerr" ]
349         then
350             # echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --""
\e[1;34m" " [K]" "\e[0m" "Cash/Check""\r"
351             echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --""\e
[1;34m" " [K]" "\e[0m" "Cash/Check" " --""\e[1;31m" " [I]" "\e[0m" "Invoice"" \r"
352             fi
353             # echo -e "\nCash"
354             paytype=Cash
355             break
356             ;;
357
358             "C" | "c")
359             if [ "$keyerr" ]
360             then
361                 # echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --""
\e[1;34m" " [K]" "\e[0m" "Cash/Check""\r"
362                 echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --""\e
[1;34m" " [K]" "\e[0m" "Cash/Check" " --""\e[1;31m" " [I]" "\e[0m" "Invoice"" \r"
363                 fi
364                 # echo -e "\nCredit"
365                 paytype=Credit
366                 break
367                 ;;
368
369             "I" | "i")
370             if [ "$keyerr" ]
371             then
372                 # echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --""
\e[1;34m" " [K]" "\e[0m" "Cash/Check""\r"
373                 echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --""\e
[1;34m" " [K]" "\e[0m" "Cash/Check" " --""\e[1;31m" " [I]" "\e[0m" "Invoice"" \r"
374                 fi

```

```

375     # echo -e "\nCredit"
376     paytype=Invoice
377     break
378     ;;
379
380     " ")
381     stty sane
382     echo -e "\e[0m"
383     clear
384     echo -e "Exited on Signal 2. Transaction" "\e[1;31m" "ABORTED." "\e[0m"
385     stty erase
386     exit 99
387     ;;
388
389     " ")
390     keyerr=1
391     echo -e -n "\e[0m"
392     # echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --"
"\e[1;34m" " [K]" "\e[0m" "Cash/Check""\r"
393     echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --""\
e[1;34m" " [K]" "\e[0m" "Cash/Check" " --""\e[1;31m" " [I]" "\e[0m" "Invoice" "\r"
"
394     ;;
395
396     *)
397     keyerr=1
398     echo -e -n "\e[0m"
399     # echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --""
\e[1;34m" " [K]" "\e[0m" "Cash/Check""\r"
400     echo -e -n " Enter Correct Type: ""\e[1;33m""[C]" "\e[0m" "Credit" " --""\e
[1;34m" " [K]" "\e[0m" "Cash/Check" " --""\e[1;31m" " [I]" "\e[0m" "Invoice" "\r"
401     ;;
402     esac
403     echo -n -e "\e[0m"
404     done
405
406     taxabletotal=`echo "$taxabletotal 1.065" | awk '{ sum = $1 * $2; printf "%.2f",sum
}'`
407     grandtotal=`echo "$taxabletotal $servicetotal" | awk '{ sum = $1 + $2; printf "%.2f
",sum }'`
408
409     while :
410     do
411     echo -e "\n"${paytype}
412
413     if [ "$paytype" = "Cash" ]
414     then
415         echo "servicetotal ="$servicetotal
416         echo "taxabletotal ="$taxabletotal
417         echo "Grand Total is:"$grandtotal
418
419         echo -n -e "Amount Tendered: $"
420         read tendered
421
422         echo "${tendered}" | sed -e 's/\./\./' | grep -q [^[:digit:]]
423
424         if [ $? = 0 ]
425         then
426             echo -e -n "ERROR:\e[1;31m Non Numeric Value Given.\e[0m Re-"
tendered=""
427
428             # foo=`echo "${tendered}" | awk '{ printf "%.2f",$1 }'`
429             # echo "foo: "$foo
430
431         elif [ "$tendered" ]
432         then
433             foo=`echo "${tendered}" | awk '{ printf "%.2f",$1 }'`
434
435             if [ "0`echo $foo | tr -d "."`" -lt "0`echo $grandtotal | tr -d "."`"
436

```

```

`" ]
437         then
438             # echo $foo-$grandtotal
439             echo -e -n "ERROR:\e[1;31m Not Enough Money. \e[0m Re-"
440             tendered=""
441         fi
442     fi
443
444     while [ ! "${tendered}" ]
445     do
446         echo -n "Enter Amount Tendered: $"
447         read tendered
448         if [ ! "${tendered}" ]
449         then
450             continue
451         else
452             echo "${tendered}" | sed -e 's/\./\./' | grep
453             if [ $? = 0 ]
454             then
455                 echo -e -n "ERROR:\e[1;31m Non Numeric Valu
e Given.\e[0m Re-"
456                 tendered=""
457                 continue
458             fi
459             tendered=`echo "${tendered}" | awk '{ print
f "%.2f", $1 }'`
460             echo "Tendered: "$tendered
461             if [ "$tendered" ] && [ "`echo $tendered |
tr -d "."`" -lt "`echo $grandtotal | tr -d "."`" ]
462             then
463                 echo -e -n "ERROR:\e[1;31m Not Enough Money
. \e[0m Re-"
464                 tendered=""
465                 fi
466             done
467             fi
468             echo
469             tendered=`echo "${tendered}" | awk '{ printf "%.2f", $1 }'`
470             echo "Grand Total: "$grandtotal
471             echo "Tendered: "$tendered
472             if [ "`echo $tendered | tr -d "."`" -ge "`echo $grandtotal | tr -d "."`" ]
473             then
474                 change=`echo $tendered $grandtotal | awk '{ diff = $1 - $2; printf
("%.2f",diff) }'`
475                 echo "Change: "$change
476                 status=done
477             fi
478             ~/bin/clickopen 1> /dev/null
479             mysql pos -e "update posdata set revenue = '$grandtotal' where job_no like
'$jobno' ;"
480             sed -i -e "s/Total Due: $          /Total Paid: $ $grandtotal/" $woapdf
481         fi
482     #End of Cash Processing
483
484     if [ "$paytype" = "Invoice" ]
485     then
486         echo "servicetotal ="$servicetotal
487         echo "taxabletotal ="$taxabletotal
488         echo "Grand Total is:"$grandtotal
489         echo "Invoice # ${invoiceno} Being Generated For: ${name}"

```

```

498
499         status=done
500
501         ~/bin/clickopen 1> /dev/null
502         mysql pos -e "update posdata set revenue = '$grandtotal' where job_no like
'$jobno' ;"
503         sed -i -e "s/Total Due: $          /Total Due: $ $grandtotal/" $woapdf
504     fi
505     #End of Invoice Processing
506
507
508
509     if [ "$paytype" = "Credit" ]
510     then
511
512         echo "servicetotal ="$servicetotal
513         echo "taxabletotal ="$taxabletotal
514         echo "Grand Total is:"$grandtotal
515
516         echo "Enter Additional Info If Any: "
517         read descr
518
519         echo
520
521         echo "Swipe Card: "
522         stty -echo
523         stty eof " "
524         read x_track1
525         read x_track2
526         stty echo
527         stty eof " "
528         x_track1=`echo $x_track1 | tr -d "%" | tr -d "?" | tr -d "\n"`
529
530         echo
531         echo $x_track1 > /tmp/x_track1
532
533
534
535
536     ## Setup For Test Card Here Both Authorize and Fail
537         # echo "card processed here"
538         result=`wget --no-check-certificate -O - "https://cardpresent.authorize.net
t/gateway/transact.dll?x_cpversion=1.0&x_login=auth.net-login&x_market_type=2&x_dev
ice_type=1&x_amount=${grandtotal}USD&x_tran_key=auth.net-trankey&x_track1=$x_track1
" 2> /dev/null`
539         responsecode=`echo $result | sed -e 's/></\n/g' | grep ResponseCode | awk -
F\> '{ print $2 }' | awk -F\< '{ print $1 }`
540         ## Put Decline, etc handler Here
541         if [ -z "$responsecode" ] || [ $responsecode -ne 1 ]
542         then
543             echo -e "Transaction " "\e[1;31m" "FAILED" "\e[0m"
544             echo "Try Card Again? "
545             while :
546             do
547                 echo -n -e "\e[0;30m"
548                 stty raw
549                 retrycard=`dd bs=1 count=1 2> /dev/null`
550                 stty -raw
551                 echo -n -e "\e[0m"
552
553                 case "$retrycard" in
554                     "n" | "N")
555                         if [ "$keyerr" ]
556                         then
557                             echo -e -n " Enter ""\e[1;33m""[Y\y]" "\e[0m" "-" " :""\e[1
;34m" " [N/n]" "\e[0m" "-" ""r"
558                             fi
559                             # echo -e "\nCASH"
560                             paytype=Cash

```

```

561         done=cash
562         break
563         ;;
564
565         "y" | "Y")
566         if [ "$keyerr" ]
567         then
568             echo -e -n " Enter ""\e[1;33m""[Y\y]" "\e[0m" "-" " :""\e[1
;34m" " [N/n]" "\e[0m" "-" "\r"
569         fi
570         # echo -e "\nCcredit"
571         paytype=Credit
572         done=redo
573         break
574         ;;
575
576         " ")
577         stty sane
578         echo -e "\e[0m"
579         clear
580         echo -e "Exited on Signal 2. Transaction" "\e[1;31m" "ABORT
ED." "\e[0m"
581         stty erase
582         exit 99
583         ;;
584
585         " ")
586         keyerr=1
587         echo -e -n "\e[0m"
588         echo -e -n " Enter ""\e[1;33m""[Y\y]" "\e[0m" "-" " :""\e[
1;34m" " [N/n]" "\e[0m" "-" "\r"
589         ;;
590
591         *)
592         keyerr=1
593         echo -e -n "\e[0m"
594         echo -e -n " Enter ""\e[1;33m""[Y\y]" "\e[0m" "-" " :""\e[1
;34m" " [N/n]" "\e[0m" "-" "\r"
595         ;;
596         esac
597         echo -n -e "\e[0m"
598         done
599     fi
600
601     if [ "$done" = "redo" ]
602 then
603         echo "redo"
604         paytype=Credit
605         done=
606         continue
607     elif [ "$done" = "cash" ]
608 then
609         continue
610     fi
611
612     echo > /tmp/response
613
614         # for f in `seq 1 2`
615         # do
616         #     echo "-" >> /tmp/response
617         # done
618     echo "Credit Card Receipt For Boca Computer Center Job Number: "$jobno >> /
tmp/response
619     # echo >> /tmp/response
620     echo $descr >> /tmp/response
621     # echo >> /tmp/response
622
623     echo "Amount: $"$grandtotal >> /tmp/response
624

```

```

625         echo >> /tmp/response
626
627         echo $result > /tmp/result
628
629 # echo $result | sed -e "s/<\\//\n          end /" | tr -d "<" | tr -s ">" " " >> /tmp
/response
630         cat /tmp/result | sed -e "s/</>\n</g" | sed -e "s/<\\//\n          end /" | tr
-d "<" | tr ">" " " | tr -s "\n" | grep -A2 ^AuthCode >> /tmp/response
631
632         # echo >> /tmp/response
633         cat /tmp/result | sed -e "s/</>\n</g" | sed -e "s/<\\//\n          end /" | tr
-d "<" | tr ">" " " | tr -s "\n" | grep -A1 ^TransID >> /tmp/response
634
635         # echo >> /tmp/response
636         cat /tmp/result | sed -e "s/</>\n</g" | sed -e "s/<\\//\n          end /" | tr
-d "<" | tr ">" " " | tr -s "\n" | grep -A2 ^Description >> /tmp/response
637
638
639
640
641 cat /tmp/response >> invoice.$jobno
642
643         echo
644
645
646         x_track1="null"
647
648         # nail -s "Another Credit Card Charge For Boca Computer Center" -r no-reply
@bocacomputercenter.com no-reply@bocacomputercenter.com < /tmp/response
649
650
651         mysql pos -e "update posdata set cc_response = '$result' where job_no =
'$jobno'"
652         mysql pos -e "update posdata set revenue = '$grandtotal' where job_no like
'$jobno' ;"
653         sed -i -e "s/Total Due: $          /Total Paid: $ ${grandtotal}/" $woapdf
654 status=done
655 fi
656 #End of Credit Processing
657 if [ "$status" = "done" ]
658 then
659         echo "done"
660         break
661 fi
662 done
663 #### end Payment Handling
664
665
666 # paginate description and enter it ino
667 # body of form
668 linecount=`cat invoice.$jobno | wc -l`
669 pagecount=`expr $linecount / 32`
670 pagecount=`expr $pagecount + 1`
671
672 startline=1
673
674 for page in `seq 1 $pagecount`
675 do
676         # setup pages
677         prnpage=$page
678         num=0000000${page}
679         len=`echo $num | wc -c`
680         digs=`expr $len - 4`
681         page=`echo $num | cut -c ${digs}-`
682
683         echo "Page $prnpage of $pagecount"
684
685
686         cp $woapdf $woapdf.$jobno.$page.ps

```

```

687
688
689     if [ "$prnpage" -eq "$pagecount" ]
690     then
691     {
692         sed -i -e "s/Total Due: $          /Total Due: $ $revenue/" $woapdf.$
jobno.$page.ps
693     }
694     elif [ "$prnpage" -lt "$pagecount" ]
695     then
696     {
697         sed -i -e "s/Thank You -- Boca Computer Center./
/" $woapdf.$jobno.$page.ps
698         # sed -i -e "s/Total Paid: $          /Continued On Next Page... /" $w
oapdf.$jobno.$page.ps
699         sed -i -e "s/Total Paid: $ $grandtotal/Continued On Next Page... /"
$woapdf.$jobno.$page.ps
700         sed -i -e "s/Total Due: $ $grandtotal/Continued On Next Page... /"
$woapdf.$jobno.$page.ps
701     }
702     fi
703
704
705
706     lastline=`expr $startline + 30`
707
708     sed -n "${startline},${lastline}p" invoice.$jobno > invoice.$jobno.$page
709
710
711     descline=1
712
713     sed -i -e "211s/(          ) 7650 X/( Page: $prnpage of $pagecount) 100
0 X/" $woapdf.$jobno.$page.ps
714
715
716
717
718     # In each page, place page from description
719     for lineno in `cat ${etcdir}lines.template.body | awk -F\: '{ print $1 }' |
head -n 32`
720
721     do
722
723         echo -n $lineno
724         linelen=`sed -n "${descline}p" invoice.$jobno.$page | wc -c`
725         psdata=`sed -n "${descline}p" invoice.$jobno.$page`
726
727         pslength=`expr $linelen \* 100`
728         sed -i -e "${lineno}s/(          ) 7650 X/( $psdata ) $pslength
X/" $woapdf.$jobno.$page.ps
729
730
731         echo "    "$descline"    "$pslength
732
733
734         descline=`expr $descline + 1`
735
736     done
737
738
739
740
741
742     startline=`expr $lastline + 1`
743
744     ps2pdf $woapdf.$jobno.$page.ps > ${logdir}inv.$jobno.$page.out 2>&1
745
746
747     cat $woapdf.$jobno.$page.ps >> $woa_form.$jobno.ps

```

```

748
749
750 cp $woapdf.$jobno.$page.ps $woaemail.$jobno.$page.ps
751 sed -i -e '121s/.*/(Boca Computer Center) 3632 X/' $woaemail.$jobno.$page.ps
752 sed -i -e '127s/.*/(160 W Camino Real #191) 3050 X/' $woaemail.$jobno.$page.ps
753 sed -i -e '133s/.*/(Boca Raton, Fl. 33432) 3447 X/' $woaemail.$jobno.$page.ps
754 sed -i -e '140s/.*/(561.961.8185) 1530 X/' $woaemail.$jobno.$page.ps
755 sed -i -e '147s/.*/(Customer Info:) 1187 X/' $woaemail.$jobno.$page.ps
756 sed -i -e "186s/.*/(Job No: $jobno ) 1375 X/" $woaemail.$jobno.$page.ps
757
758 cat $woaemail.$jobno.$page.ps >> $woa_all.$jobno.ps
759 # ps2pdf $woaemail.$jobno.$page.ps >> ${logdir}inv.$jobno.$page.out 2>&1
760
761 mv $woapdf.$jobno.$page.ps $logdir
762 mv $woaemail.$jobno.$page.ps $logdir
763 mv $woapdf.$jobno.$page.pdf $outdir
764 # mv $woaemail.$jobno.$page.pdf $outdir
765 mv invoice.$jobno.$page $logdir
766
767 description=`cat invoice.$jobno | tr -d "&" | tr -d "' "`
768
769 done
770 ##---End Pagination Routine
771
772
773
774
775
776 mv invoice.$jobno $logdir
777 mv $woapdf ${logdir}$woapdf.$jobno
778 # mv $woaemail ${logdir}$woaemail.$jobno
779
780 ps2pdf $woa_all.$jobno.ps > /dev/null 2>&1
781 ps2pdf $woa_form.$jobno.ps > /dev/null 2>&1
782 # xpdf -geometry 800x700+200+0 $woa_full.$jobno.pdf 2> /dev/null
783
784
785 timeout 41 xpdf -ps "|lpr -#2 -PInvoice Invoice.print.$jobno.pdf" -geometry 800x70
0+200+0 $woa_all.$jobno.pdf 2> /dev/null
786
787
788 mv $woa_all.$jobno.ps $logdir
789 mv $woa_all.$jobno.pdf $outdir
790
791
792 mv $woa_form.$jobno.ps $logdir
793 mv $woa_form.$jobno.pdf $outdir
794
795
796
797 # for f in ${outdir}$woapdf.$jobno.$page.pdf
798 # do
799 # xpdf $f
800 # xpdf -geometry 800x700+200+0 $f 2> /dev/null
801 # done
802
803
804
805 echo -e "JobNo: $jobno\nAmount: $grandtotal" | nail -s "Another Invoice Generated F
or Boca Computer Center" -r no-reply@bocacomputercenter.com no-reply@bocacomputerce
nter.com
806
807 # echo -e "JobNo: $jobno\nAmount: $revenue" | nail -s "Another Invoice Generated Fo
r Boca Computer Center" -r no-reply@bocacomputercenter.com bocacomputercenter@bocac
omputercenter.com
808
809
810
811 invoice_datetime=`date +%Y-%m-%d" "%H:%M:%S`

```

```
812
813
814 mysql pos -e "update posdata set invoice_datetime = '$invoice_datetime' where job
_no = '$jobno'"
815 mysql pos -e "update posdata set status = 'Closed' where job_no = '$jobno'"
816
817 if [ "$paytype" = "Invoice" ]
818 then
819 mysql pos -e "update posdata set status = 'Pending' where job_no = '$jobno'"
820 fi
821
822
823
```