SDC: Software Development Cheatsheets

Last update May 15, 2015

RSA

Contents

1

PHP

2

Perl

3 MySQL

3.1 Configuration

3.1.1 Config file

Go through this ritual a few thousand times over the course of a year, and you've just lost several hours of time due to merely logging in. Eliminate this tedious step by creating a file named .my.cnf and placing it in your home directory.

```
vim /home/rezasa/.my.cnf
```

3.1.2 Enable tab compilation

Add the following to the configuration file.

Listing 3.1 : Enable tab compilation

```
[mysql]
auto-rehash
```

3.1.3 Silent mode

If you produce output from mysql to be used with this program, a header row would throw off the results because summarize would count the column heading. To create output that contains only data values, suppress the column header row with the silent mode.

Listing 3.2 : Enable silent mode

```
[mysql]
silent
```

3.1.4 Automatically Switch to a Database

Listing 3.3 : Automatically Switch to a Database

```
[client]
database = your\_database\_name
```

3.2 Commands

3.2.1 Database structure

Listing 3.4 : Database manipulation and query commands

```
use [database name]
show databases;
show tables;
desc [table-name];
show index from table_name
```

3.2.2 Table Commands

Create Table

Listing 3.5 : Create Table

```
CREATE TABLE table_name (create_clause1, create_clause2, ...)
```

Creates a table with columns as indicated in the create clauses.

- create_clause
  - column name followed by column type, followed optionally by modifiers. For example, `"gene_id INT AUTO_INCREMENT PRIMARY KEY"` (without the quotes) creates a column of type integer with the modifiers described below.
- create_clause modifiers
  - **AUTO_INCREMENT** : each data record is assigned the next sequential number when it is given a NULL value.
  - **PRIMARY KEY** : Items in this column have unique names, and the table is indexed automatically based on this column. One column must be the PRIMARY KEY, and only one column may be the
  - **PRIMARY KEY** : This column should also be NOT NULL.
  - **NOT NULL** : No NULL values are allowed in this column: a NULL generates an error message as the data is inserted into the table.
  - **DEFAULT** value : If a NULL value is used in the data for this column, the default value is entered instead.

Drop Table

Listing 3.6 : Drop Table

```
DROP TABLE table_name
```

Removes the table from the database. Permanently! So be careful with this command!

ALTER TABLE

Listing 3.7 : ALTER TABLE

```
ALTER TABLE table_name ADD (create_clause1, create_clause2, ...)
```

Adds the listed columns to the table.

ALTER TABLE ... DROP ...

Listing 3.8 : ALTER TABLE ... DROP ...

```
ALTER TABLE table_name DROP column_name
```

Drops the listed columns from the table.

Drop Table

Listing 3.9 : Drop Table

```
DROP TABLE table_name
```

Removes the table from the database. Permanently! So be careful with this command!

3.2.3 Data Commands

INSERT

Listing 3.10 : INSERT table VALUES (...)

```
INSERT [INTO] table_name VALUES (value1, value2, ...)
```

Insert a complete row of data, giving a value (or NULL) for every column in the proper order.

Listing 3.11 : INSERT table (...) VALUES (...)

```
INSERT [INTO] table_name (column_name1, column_name2, ...) VALUES (value1, value2, ...)
```

Listing 3.12 : INSERT table SET Column

```
INSERT [INTO] table_name SET column_name1=value1, column_name2=value2, ...
```

Insert data into the listed columns only. Alternate forms, with the SET form showing column assignments more explicitly.

DELETE

Listing 3.13 : DELETE FROM tablename

```
DELETE FROM table_name WHERE where_clause
```

UPDATE

Listing 3.14 : UPDATE table SET column

```
UPDATE table_name SET column_name1=value1, column_name2=value2, ... [WHERE where_clause]
```

Alters the data within a column based on the conditions in the where_clause.

3.2.4 Other commands

See http://www.bios.niu.edu/johns/bioinform/mysql_commands.htm.

3.3 Tips

3.3.1 Shell Alias

You can use shell aliases to execute a specific command:

```
alias usrcount="mysql -u appadmin -p myapplication -e "select count(id)from users""
```

3.3.2 Formating Output

Rendering query results in a vertical format:

```
SELECT * from users WHERE id=1\G;
```

Or without semicolon using:

```
SELECT * from users WHERE id=1\g
```

MySQL queries can be formatted in a better way using less:

```
\P less -S
```

you don't have to look at wrapped rows, you can just scroll right and left

```
pager column -t less -S|
```

Or you can use vim

```
pager vim -R -c "set nowrap" -
```

Or use default pager `nopager or n`

4 Java

4.1 Java

Listing 4.1 : Embeddable

```java
@Entity
public class ReportCostEntity implements  Serializable {

  @Id
  private Long id;

  @Embedded
  @AttributeOverrides( {
    @AttributeOverride(name="coveredByGrant.amount", column = @Column(name="
        contracted_coveredByGrant") ),
    @AttributeOverride(name="foundedFromOwnResources.amount", column = @Column(name="
        contracted_foundedFromOwnResources")),
    @AttributeOverride(name="personalContribution.amount", column = @Column(name="
        contracted_personalContribution"))
  } )
  private ReportCostValues contracted;

  @Embedded
  @AttributeOverrides( {
    @AttributeOverride(name="coveredByGrant.amount", column = @Column(name="
        current_coveredByGrant") ),
    @AttributeOverride(name="foundedFromOwnResources.amount", column = @Column(name="
        current_foundedFromOwnResources")),
    @AttributeOverride(name="personalContribution.amount", column = @Column(name="
        current_personalContribution"))
  } )
  private ReportCostValues current;

  @Embedded
  @AttributeOverrides( {
    @AttributeOverride(name="coveredByGrant.amount", column = @Column(name="
        previousReport_coveredByGrant") ),
    @AttributeOverride(name="foundedFromOwnResources.amount", column = @Column(name="
        previousReport_foundedFromOwnResources")),
    @AttributeOverride(name="personalContribution.amount", column = @Column(name="
        previousReport_personalContribution"))
  } )
  private ReportCostValues previousReport;
}
```

How maven works?

- There are different settings that should be modified locally in-order for goals to work. For example the following items should be added to `~/.m2/settings.xml` file:

Listing 4.2 : profile

```
<profiles>
  <profile>
    <id>glassfish-profile</id>
    <properties>
      <glassfish.user>admin</glassfish.user>
      <glassfish.password>nassim</glassfish.password>
      <glassfish.domain.host>gammer.se.axis.com</glassfish.domain.host>
      <glassfish.local.directory>/Users/rezasa/glassfish4</glassfish.local.directory>
      <glassfish.domain.name>domain1</glassfish.domain.name>
      <glassfish.domain.httpPort>8080</glassfish.domain.httpPort>
      <glassfish.domain.adminPort>4848</glassfish.domain.adminPort>
      <migrate_db_url>jdbc:mysql://gammer.se.axis.com:3306/crm_test</migrate_db_url>
      <migrate_db_user>pos_test</migrate_db_user>
      <migrate_db_user_password>pos_test</migrate_db_user_password>
    </properties>
  </profile>
</profiles>
```

- It is important for some commands to be in the correct directory and then execute them. For example in `pos-services`. A list of known commands to me are:

  - `mvn clean test`
    Then you go to for example `pos-services/target/surefire-reports` or `pos-services/target/jacoco-u`
    to view the reports and see the coverage.
  - `mvn exec:exec -P glassfish-profile,db-migrate`
    Use this in `pos-services` folder to do database migration
  - Another possibility is to recompile and redeploying
    `mvn compiler:compile glassfish:redeploy -P glassfish-profile`

4.1.1 Resources

In order to access a resource we can use:

Listing 4.3 : Accessing a resource through bundle:

```
ResourceBundle messageBundle = ResourceBundle.getBundle("ValidationMessages");
return messageBundle.getString(messageId);
```

In order that the resource would be available in both Test and the Production we should put them in `src/main/resources`. For example for the above example if we put `ValidationMessages.properties` in the specified folder it would be accessible in both unit tests and production server.

4.1.2 Unit Testing

4.1.3 String.format

Samples:

```
    String formattedString = String.format("Order with OrdId : %d and Amount: %d is missing"
        , 40021, 3000);
    System.out.printf("Amount : %08d %n" , 221);

    System.out.printf("positive number : +%d %n", 1534632142);
    System.out.printf("negative number : -%d %n", 989899);

    //printing floating point number with System.format()
    System.out.printf("%f %n", Math.E);

    //3 digit after decimal point
    System.out.printf("%.3f %n", Math.E);

    //8 charcter in width and 3 digit after decimal point
    System.out.printf("%8.3f %n", Math.E);

    //adding comma into long numbers
    System.out.printf("Total %,d messages processed today", 10000000);
```

4.2 Maven

Inorder to `M2_HOME` be accessible through all applications in MacOsX you should configure `lunchd`. `launchctl setenv M2_HOME /usr/local/apache-maven/apache-maven-3.2.3`

4.3 glassfish

4.3.1 List of glassfish subcommands

Here you can find a list of asadmin subcommands (you can sort by package for better grouping). Some of the important ones are:

- change-admin-password
- login
- rotate-log

4.3.2 Save admin login

You can change the admin password using: `asadmin change-admin-password`. To save the password and bypass the authentication every time you need to login one time using: `asadmin login`

4.4 Android

4.4.1 Gradle

Solving dependency problems when building

- To show the dependencies: `gradle -q dependencies`. Also see this link to learn how to solve the problem with dependencies. **Important**: to exclude a library you should use:

Listing 4.4 : Gradle compile line

```
compile ('com.foursquare:foursquare-android-oauth:1.0.1'){
  exclude module: 'support-v4'
}
```

4.5 JPA

- In `OneToMany` relationships, be sure to remove `@NotNull()` constraint from the child table! (very important)
- The entity should be added to `/Users/rezasa/git/ebizz-pos-api-java/pos-services/src/main/resources/`

5 Rest APIS

5.1 Resource Naming

- **Pluralization**:
  There are good arguments on both sides (pluralizers and the "singularizers"), but the commonly-accepted practice is to always use plurals in node names to keep your API URIs consistent across all HTTP methods. The reasoning is based on the concept that customers are a collection within the service suite and the ID (e.g. 33245) refers to one of those customers. Using this rule, an example multi-node URI using pluralization would look like (emphasis added):

  GET `http://www.example.com/customers/33245/orders/8769/lineitems/1`

  However, there are cases that it doesn't make sense. like:
  GET|PUT|DELETE `http://www.example.com/configuration`

- **The official rules for URIs:**
  See Internet Engineering Task Force (IETF) - URI-Template

6 Javascript

6.1 JavaScript

Good References

- A very good reference for learning Closures
- Using closures in a sample event

Listing 6.1 : Using closures in a sample event

```
//todo: add the above section to the hover only for anchor elements
addEventListener('mousemove', this.on_mousemove(this), false);

 };

 extractor.prototype.on_mousemove = function (scope) {
   return function (e) {
     var that = scope;
     var srcElement = e.srcElement;
     if (that.prevDOM == null || !srcElement.isSameNode(that.prevDOM)) {
       var anchor = extractor.findAnchorElement(srcElement);
       if (anchor != null) {
         var hRef = anchor.attr('href');
         if (that.prevHRef != hRef) {
           that.doMouseOut();
           that.doMouseIn(anchor, hRef);
         }
       }
       else {
         that.doMouseOut();
       }
     }
     that.prevDOM = srcElement;
   }
 };
```

6.2 Angular

Angular main features are available through

- **ng-app** The angular application
  First parameter is the name of the module and the second is the list of dependencies(injectables).

```
// the setter module
angular.module('myApp', []);
// this method fetches the app
angular.module('myApp')
```

- ng-controller
  A typical controller could be like this

Listing 6.2 : controller

```
function MyController($scope) {
  var updateClock = function () {
    $scope.clock = new Date();
  };
  setInterval(function () {
    $scope.$apply(updateClock);
  }, 1000);
  updateClock();
};
```

- **Scopes**

  - he glues between application and the models (source of truth for application state)
  - propagate changes through `apply` mechanism
  - propagate events?
  - application logic could be in controller and working data in the scope of controller.
  - The `ng-app` is binded to the `$rootscope` .

6.3 (Ocean)

`mysql -u posapp -p` `rwqfam6cxj`

6.4 JQuery

Selectors

- A tag name: `$('p')` gets all paragraphs in the document.
- An ID: `$('#some-id')` gets the single element in the document that has the corresponding some-id ID.
- A class: `$('.some-class')` gets all elements in the document that have a class of some-class.

Listing 6.3 : Change an element onload

```
$(document).ready(function() {
  $('.poem-stanza').addClass('emphasized');
});
```

Check if an element is inside another specific element :

```
if ( $(".child-element").parents("#main-nav").length == 1 ) {
  // YES, the child element is inside the parent
} else {
  // NO, it is not inside
}
```

Find if there is a relation between parent and child :

```
   if($(' #childElementID' , parentElementId).length == 1) // child is inside parent
```

Find a certain element within the children :

```
    $(".element")find("a").first();
```

Get element's attribute :

```
    $(".element").attr('href')
```

Change element Style :

```
    $(".element").css("background-color", "blue");
```

Get element text :

```
  $(".element").text();
```

6.5 NodeJS

Where is nodejs local directory ?

- use `npm config ls -l` to get the list of all configurations.
- Look for "prefix" item.

6.5.1 RoboHydra

Somehow RoboHydra doesn't now its plugin folder so it should be provided with the command and the plugin should be copied or linked to that folder:

```
robohydra -n -I /usr/local/lib/node_modules/robohydra/plugins/ -P pos
```

6.6 Linux

6.6.1 ZSH

Shortcuts:

```
        Ctrl+X u             ->   undo
        echo $PWD<TAB>       ->  echo /home/pws/zsh/projects/zshguide
        echo !!<TAB>         ->  echo echo 3.1.7
        echo ~/.z*<TAB>        ->  echo /home/pws/.zcompdump /home/pws/.zlogout ↙
                            ↪ /home/pws/.zshenv /home/pws/.zshrc
        print ${ZSH_V<TAB>   ->   ${ZSH_VERSION}
```

6.6.2 Autoload

autoload can import function definition header (but not the body) so that the functions are loaded once they are used.

To see the functions that are currently loaded via autoload, you can use:
`autoload -U | less`

Autoload functions are located in "`/usr/share/zsh/functions`". A common pattern for loading these functions is `autoload -U colors &&colors`. The colors script is located in `/usr/share/zsh/functions/Misc`.

6.6.3 Colors

One safe way to use colors is defining them via `tput`.

```
BLACK=$(tput setaf 0)
RED=$(tput setaf 1)
GREEN=$(tput setaf 2)
YELLOW=$(tput setaf 3)
LIME_YELLOW=$(tput setaf 190)
POWDER_BLUE=$(tput setaf 153)
BLUE=$(tput setaf 4)
MAGENTA=$(tput setaf 5)
CYAN=$(tput setaf 6)
WHITE=$(tput setaf 7)
BRIGHT=$(tput bold)
NORMAL=$(tput sgr0)
BLINK=$(tput blink)
REVERSE=$(tput smso)
UNDERLINE=$(tput smul)
```

the other way is by loading colors script. `autoload -U colors &&colors`
and then using them in code like: `echo $bg[red]$fg[white] reza $reset_color`

Inorder to use these colors you can use the following shortcuts:

```
$fg_bold[color]
$fg_no_bold[color]
$bg[color]
$bg_bold[color]
$bg_no_bold[color]
$boldcolor
$resetcolor
```

Or use it in prompt like:

```
PS1="%{$fg[red]%}%n%{$reset_color%}
```

To use other features like underline, overline, etc. you should do a little more:

```
lc=$'\e[' rc=m    # Standard ANSI terminal escape

$fg_bold[red] reza $fg_no_bold[red]shams "$lc"09";${color[white]}$rc" amiri $reset_color
```

Listing 6.4 : The variables that could be used with color

```
color=(
    # Codes listed in this array are from ECMA-48, Section 8.3.117, p. 61.
    # Those that are commented out are not widely supported or aren’t closely
    # enough related to color manipulation, but are included for completeness.

    # Attribute codes:
    00 none                 # 20 gothic
    01 bold                 # 21 double-underline
    02 faint                  22 normal
    03 standout               23 no-standout
    04 underline              24 no-underline
    05 blink                  25 no-blink
    # 06 fast-blink           # 26 proportional
    07 reverse                27 no-reverse
    08 conceal                28 no-conceal
    # 09 strikethrough        # 29 no-strikethrough

    # Font selection:
    # 10 font-default
    # 11 font-first
    # 12 font-second
    # 13 font-third
    # 14 font-fourth
    # 15 font-fifth
    # 16 font-sixth
    # 17 font-seventh
    # 18 font-eighth
    # 19 font-ninth

    # Text color codes:
    30 black                  40 bg-black
    31 red                    41 bg-red
    32 green                  42 bg-green
    33 yellow                 43 bg-yellow
    34 blue                   44 bg-blue
    35 magenta                45 bg-magenta
    36 cyan                   46 bg-cyan
    37 white                  47 bg-white
    # 38 iso-8316-6           # 48 bg-iso-8316-6
    39 default                49 bg-default

    # Other codes:
    # 50 no-proportional
    # 51 border-rectangle
    # 52 border-circle
    # 53 overline
    # 54 no-border
    # 55 no-overline
    # 56 through 59 reserved
)
```

6.6.4 alias

To remove an alias, the unalias command is used, like so: `unalias foo`

6.6.5 Super user setup

To have a *login shell session* you should use: `su -` or `su -l`. This means that the user's environment is loaded and the working directory is changed to the user's home directory.

6.6.6 check to see if a function exists

```
if [ type __promptline \&>/dev/null ]; then
        __promptline
fi
```

6.7 Commands

6.7.1 chown

In order to change the owner of a file or directory to a user or group which have special characters like space you could use the following workaround: `id reza`

Listing 6.5 : Process Tree

```
id reza


uid=689677042(rezasa) gid=868142252(AXISNET\Domain Users) ...

sudo chown -R rezasa:868142252 ../spec
```

6.7.2 xwininfo

Can give you geometrical size of the current window.

6.8 Linux Processes

6.8.1 Getting ProcessInfo

Listing 6.6 : Getting pid

```
pidof httpd
pidof apache2
pidof firefox
```

Listing 6.7 : Getting all pids

```
ps aux | grep httpd
ps aux | grep apache2
ps aux | grep  firefox

pgrep firefox
# list the process called sshd which is owned root
pgrep -u root sshd
```

Listing 6.8 : Process Tree

```
# To display a tree of processes
pstree
# Print a process tree using ps
ps -ejH
ps axjf
```

Listing 6.9 : Get SecurityInfo

```
ps -eo euser,ruser,suser,fuser,f,comm,label
ps axZ
ps -eM
```

6.8.2 Killing processes

Listing 6.10 : Kill command

```
kill [signal] PID
kill -15 PID
kill -9 PID
kill -SIGTERM PID
kill [options] -SIGTERM PID
kill -9  pid1 pid2 pid3
```

Listing 6.11 : KillAll command

```
killall Process-Name-Here
killall -15 lighttpd
killall -9 lighttpd
killall -9 firefox-bin
```

6.8.3 htop

htop is interactive process viewer just like top, but allows to scroll the list vertically and horizontally to see all processes and their full command lines. Tasks related to processes (killing, renicing) can be done without entering their PIDs.

Listing 6.12 : installing htop

```
sudo apt-get install htop
yum install htop
htop
```

6.8.4 atop

The program **atop** is an interactive monitor to view the load on a Linux system. It shows the occupation of the most critical hardware resources (from a performance point of view) on system level, i.e. cpu, memory, disk and network. It also shows which processes are responsible for the indicated load with respect to cpu- and memory load on process level;

Listing 6.13 : atop

```
atop
```

6.9 Finding files

6.9.1 Update locate database

for locate to work use: `sudo updatedb`
See the status of locate database: `locate -S`

6.9.2 Remove located files

`locate menu_ent/menu_axis_crm_mobile.inc | xargs -ixxx echo 'xxx'`

6.10 Installing applications on Linux

6.10.1 Uncompress tar.gx

`tar -zxvf ArgoUML-0.34.tar.gz`

6.10.2 Install a deb file

`sudo dpkg -i DEB_PACKAGE`

6.10.3 Remove a deb file

`sudo dpkg -r PACKAGE_NAME`

6.10.4 Upgrade

`sudo apt-get update` `sudo apt-get upgrade` `sudo apt-get install -f`

6.10.5 Add keys

for example Spotify:
`sudo apt-key adv --keyserver hkp://keyserver.ubuntu.com:80 --recv-keys 94558F59`
This may not work so instead use: `gpg -a --export 94558F59` sudo apt-key add -| `gpg -a --export 94558F59` apt-k

6.11 Crontabs

To edit your crontabe file run: `crontab -e`

6.11.1 Setting up ssh for a remotehost

Listing 6.14 : making an alias for a remotehost

```
~/.ssh/config

 these lines to the file
 parastoo
 HostName 192.168.1.200
 User parastoo
```

Now to reduce prompting for for passphrase:

Listing 6.15 : Generating key and adding it to ssh-agent

```
ssh-keygen -t rsa

ls  . ssh

cat $HOME/.ssh/id_rsa.pub | ssh parastoo \
'cat >>  . ssh/authorized_keys && echo "copied"'

exit # exit remote computer

ssh-add
```

6.12 KDE

6.12.1 Restart Plasma on KDE

If taskbar is locked down you can use `Alt-F2` and type: `kbuildsycoca4 &&kquitapp plasma-desktop &&kstart plas`

6.12.2 Hardware

Mouse To see what button are available use the following command: `xev | grep button`

7 Git

7.1 Branches

`git checkout -b <new_branch>`
BRANCHES SHOULD ALWAYS BE IN LOWER CASE
Show all branches: `git fetch origin`

```
git branch -v -a
```

```
git checkout remotes/origin/dev/adp_online_agree_t46324
```

```
git branch --set-upstream dev/show_nfr_warnings_58187 ↙
            ↳ origin/dev/show_nfr_warnings_58187
```

```
git fetch origin
```

```
git branch -v -a
```

```
git checkout -b test origin/test
```

7.2 Aliases

Adding aliases: Basically you just need to add lines to `~/.gitconfig`

```
[alias]
st = status
ci = commit -v
Or you can use the git config alias command:
```

`git config --global alias.st status` `git config --global alias.ci 'commit -v'`
A manual alias like `gitall` could be added to see the current listing of aliases:

```
alias gitall="git config --list | grep alias"
```

7.3 Reverting

The easiest way is to: `git reset --hard` `git reset --hard origin/master`
Recovering a removed file

```
git reset HEAD file
```

```
git checkout --file
```

ignore the last commit reverts everything!

```
git reset --hard "HEAD^"
```

ignore the last commit keeps everything
`git reset --soft "HEAD^"`

shows all branches `gitk --all`

Show the changes changes `vimdiff agreement_new/new.php agreement/new.php`

```
git branch --set-upstream dev/show_nfr_warnings_58187 origin/dev/show_nfr_warnings_58187
```

`git status -s`
`git commit -a -m "message"` //will add modified files and commit
`git config --global color.ui true`
`git log --all`

7.4 Stashing

In Git you can drop your current work state in to a temporary storage area stack and then re-apply it later. The simple case is as follows:

```
stash
stash pop
stash drop stash@{1}
stash save "My stash message"
stash list
stash apply stash@{1}
stash branch                       #Create a branch from stash
stash show -p                      #Show the differences
```

```
add -i
add -p

log -p
log --stat
log --author=Andy
log --grep="Something in the message"
log lib/foo.rb
log --since=2.months.ago --until=1.day.ago
log --since=2.months.ago --until=1.day.ago --author=andy -S "something" --all-match

log master ^origin/master
log origin/master ^master

unstage lib/foo.rb

blame lib/foo.rb

reflog

mergetool --tool-help
config --global alias.lol "log --pretty=oneline --abbrev-commit --graph --decorate"
config --global alias.unstage "reset HEAD"
config --global diff.tool vimdiff
config --global difftool.prompt false
config --global alias.d difftool
```

7.5 Squashing commits with rebase

Use `git lol` to see the logs and decide how many commits you want to squash. Do a rebase by mergin for example last 4 commits.
`git rebase -i HEAD~4`
An easier and safer way would be: `git merge --squash branch`

7.6 Amending last commit

To change the last commit message:

7.7 How to fix cygwin git unnecessary change detection

Use `git config --local -e` and set `filemode=false`

```
commit --amend
commit --amend -m "New commit message"
```

7.8 How to see the origin address

`git remote show origin`

7.9 How to use a pre-written commit message

`git commit -F commit.log`

7.10 How to save password when using git (see src)

- Linux:

```
        sudo apt-get install libgnome-keyring-dev
        cd /usr/share/doc/git/contrib/credential/gnome-keyring
        sudo make
        git config --global credential.helper
        /usr/share/doc/git/contrib/credential/gnome-keyring/
        git-credential-gnome-keyring
```

- Mac:

```
        git config --global credential.helper osxkeychain
```

- Windows:
  For Windows, there is a helper called git-credential-winstore.exe or wincred in msysgit.

7.11 What if a file is no longer being tracked by git?

The only way that I found is this (keep a copy of file beforehand.)

```
git rm --cached path/to/file
git reset path/to/file
```

7.12 Working with remote branches

- Checkout a remote branch: `git co hotfix/my_fix`
- Delete a remote branch: `git push origin --delete hotfix/my_fix`

7.13 Git flow

See this document.

8 ChromeExtension Development

8.1 Setting up development environment in WebStorm

Agenda

- How can I set up Chrome for extension development?
  - WebStorm
  - Visual Studio
  - Chrome Development Tools
- What technologies are used to write extensions for Chrome?
  - node.js modules (gulp, express)

Adding stub libraries for having intellisense support

- There is a some-more finish stub record that can be total as a library to get formula completion. It's a partial of a Closure Compiler project. Download chrome_extensions.js. In WebStorm add this files to a library and use it in the project (see here). Also you need a stub for chrome API (get it here), library or stub could (be configured in WebStorm).
- ...

LiveEdit

- Use Gulp for reloading extension whenever a change occurs. Create a gulpfile.js in app folder (see A.1). Goto the extension directory and use `gulp dev`.
- Content scripts should be reloaded (see this question).

Chrome APIs References

- Web APIs
- Chrome JavaScript APIs such as (tabs, history, bookmarks, events, alarms ...)
- Document Object Mode (DOM)
- Creating a basic chrome extension
- Chrome extension samples
- Chrome app samples

- Google Chrome Development: Frequently Asked Questions
- Google Guids : Architecture
- Manifest file format
- JQuery API reference.

How to use JQuery in a chrome extension [1]

- Get a copy of JQuery and add `jquery.min.js` to the app folder.
- Add its path to manifest.json:
  `"background": "scripts": [ "scripts/jquery-2.1.3.js", ...] ,`

How to include another javascript in the current extension?

- One way is to add the script as background in manifest.json file like:
  `"background": "scripts": [ "xxx.js", ...] ,`
- Another way is to add it as content script (see this question)
- For value passing see this question and this one.

How to run a simple JQuery using browser actions?

- How to catch popup onclick event?
  - you cannot add onclick event listener with the `default_popup` use `popup` instead.
- To use JQuery in the active tab, it should be injected into the page (see this question)

How to deminify a script?

- Use jsbeautifier.org or use the referred plugins at that page.
- Chrome developer tools has this feature built-in. Bring up the developer tools (pressing F12 is one way), in the Sources tab, the bottom left bar has a set of icons. The "" icon is "Pretty print" and does this conversion on demand.

How to catch hover event in an extension?

- You can use onMouseMove event for that purpose (see this question).
- If you want to have a delay before hover (see A.2).

How to send background http request and act upon the results ?

```
        $.get(href, function (data) {
          var jq = $($.parseHTML(data));
          var res = jq.find('#profile_videos a .p_header_bottom');
          console.log(href + " : " + res.text());
        });
```

How to be notified when new contents are loaded ?

- One way is using `DOMSubtreeModified` which is complicated
- Another way is using `livereload` plugin which is not efficient.

How to inject html using content.js ?

```
window.addEventListener("load", function () {
  var url = chrome.extension.getURL("popup-info.html");
  $.get(url, function (data) {
    $('body').prepend( '<style>some style...</style>' + data )
  });
});
```

Bibliography

[1] Creating a chrome extension that uses jQuery to manipulate the DOM of a page. *Carl Topham, 2015-02-01*, (online-link)

9 Other

9.1 TexStudio

How to write scripts for TexStudio?

- see manual.

Writing Macros Go to Macros->Edit Macro

```
%SCRIPT
txt = cursor.selectedText()
editor.write("\\ic{"+txt+"}")
cursor.clearSelection()
```

Or you can use the normal with and then use different `%<defaultcode%>` as codeblock place holders:

Listing 9.1 : Using Normal Text

```
  \begin{javacode}[%<title%>]
    %<codeblock%>
  \end{javacode}
```

Finally using abbreviations you can easily recall a macro. It's important to use backslash(\) at the beginning of the abbreviation to let the command be executed.

9.2 IntelliJ

What are some of the IntelliJ's most important shortcuts ?

- Build: `ctrl + f9`
- To show tool windows press `alt(Win)|command(Mac)` twice
- To navigate to any part of the file path just press `ctrl + click`
- **Auto Competition**
  - Auto complete filename: `ctrl + SPACE` (twice first time root second time all)
  - Auto complete a word name from open files (Cyclic Expand Word): `alt + /` (do it several times to browse through)
  - Postfix completion (check on IntelliJ) after an string or number put a dot(.)
  - **fishshell**: you can use fishshell to have auto completion in IntelliJ
- **Running**

  - Toggle breakpoint: `ctrl + F8`
  - Step over: `F10`
  - Step into: `F11`
  - Resume: `F9`
  - RunTest: `ctrl + shift + F10`
  - RunTest in Debug mode: `ctrl + shift + F9`
  - RunTest menue: `alt + shift + F10`

- **Code Editing**
  - Moving up/down a block: `ctrl + alt + UP/DOWN`
  - For commenting you can use `/**`: if you want to have shortcut use `Fix Doc Comment` action
  - Multiple selection (rename): `alt + J (win/linux) ctrl + G|`
  - Insert/Show the list of **live template**: `ctrl + J`
  - Surround with: `ctrl + alt + T`
  - Toggle folding: `ctrl numpad +/-`
  - Reformat Code: `ctrl + alt + L`
  - Cycle through the history of changes position: `ctrl + shift + BACKSPACE`
  - Paste from clipboard buffer: `ctrl + shift + V`
  - Incremental Selection: `ctrl + W` (should be changed perhaps?)
  - Delete Line: `ctrl + Y`
  - Comment/Uncomment: `ctrl + /`
  - Refactor this: `ctrl + shift + R`
  - Duplicate line or block: `ctrl + D`
  - Jump between highlighted syntax errors: `F2 | shift F2`
  - Navigate through the methods of the open file: `alt UP/Down`
  - Join lines and remove unnecessary white spaces: `ctrl + shift + J`
  - When using code completion use tab instead of enter to completly replace the text
  - To see the View|Quick Definition use : `ctrl + shift + I`
  - To reorganize the code you can use on statement or class level: `ctrl + shift + UP/DOWN`
  - Generate Code : `ctrl + shift + INSERT`
  - Show Generate Menu : `alt + INSERT`
  - Implement methods : `ctrl + I`
  - **Sorround** with : `ctrl + alt + T`
  - **Refactoring**
    * Use Refactor|Copy to create a copy of the current class: `F5`
    * Use Refactor|Move to turn a method to a class or extract: `F6`
    * Use Refactor|Extract variable to refactor a variable : `ctrl + alt + V`
    * Use Refactor|Extract method to refactor a block to a method : `ctrl + alt + M` | `command + alt + M`

- **Finding**
  - **Show usage** `ctrl + alt + F7` : show usage of a member in the whole project

  - Use `ctrl + shift + F7` to search for the usage of a certain variable in the current file. Use F3 to navigate the results and double escape to get out of the highlight mode. Moreover if you put the caret on a return point of a method, all exit points would be highlighted.
  - Find member(symbol) : `ctrl + alt + shift + N`
  - Finding actions: `ctrl + shift + A` (for example type save live)
  - List class members: `ctrl + F12`
  - Goto file (Navigate File): select a text and then use `ctrl + shift + N`. **Note** if you right-click on a folder and set it as (Mark directory as->)"resource root" in manycases you can instantly jump to file by just using ctrl+click
  - Show recently changed files: `Ctrl + Shift + E`
  - Navigate to a filename: `ctrl + N`
  - Goto Line no: `ctrl + G`
  - Find all changes that happened to the project: `alt + shift + C`
  - Navigation bar is a quick alternative to project view: `alt + HOME`

- **VCS Management**
  - Show CVS changes: `alt + 9`
  - Show commit changes dialog: `ctrl + K`
  - Show push changes dialog: `ctrl + shift + K`
  - To compare two files, select them and use `ctrl + D`

Moreover, you can check the following page : IDEA Reference Card or alternately choose Help|Default Keymap Reference.

How to fix cursor movement

- Go to settings [Ctrl+Alt+S]
- Choose [Editor → Smartkeys]
- Check the [Use "CamelHumps" Words] button

How to integrate with and issue tracking system

- **youtrack**
  - The url is http://<username>.myjetbrains.com/youtrack
  - The login name is the user id used in youtrack profile (by default is root)
  - In order to only show tasks only related to the current project add the pattern to search textbox: `project:LinkQuery` (you should probably remove assign:me as well)

9.2.1 Updating Maven Repo

In order to be able to search for maven artifacts Maven Central Repository should be updated. Goto Maven Repositories and update http://repo.maven.apache.org/maven2.
When changing the maven POM file we need to reimport it in IntelliJ to be able to compile the project. This will cause **project.iml** file be updated which is needed for IntelliJ to be able to compile.

9.3 Cygwin

9.3.1 Configuration with Console2

One of the good things to use in conjunction with Cygwin is Console2 (see these discussions for normal use, Cygwin, mac iterm2, and powershell).
In order to use Cygwin shell in Console2 you should use: `cmd /c c:\cygwin64\bin\zsh -l` as the shell command for the main tab. The settings for Console2 is saved in `%APPDATA%/console`.

Installing Oh-My-ZSH on Cygwin64

- Use the following commands to install it:

  Listing 9.2 : Installing oh-my-zsh for cygwin

  ```
  wget --no-check-certificate https://raw.github.com/haithembelhaj/oh-my-cygwin/master/
      oh-my-cygwin.sh -O install.sh

  sh install.sh
  ```

- if you encounter a problem and received errors[1] when executing zsh, use the following remedy:

  ```
  rm -f ~/.zcompdump; compinit
  ```

  now noticed that .zcompdump was named .zcompdump-modhelius-dell-5.0.2 and this file was almost empty. After running compinit I noticed that a file only named .zcompdump was created, taking the content of this file and copying it into .zcompdump-modhelius-dell-5.0.2 fixed the problem for me[reference].

Setting up colors in Console2 To use Solarized color theme with Console2, edit `%APPDATA%/console/console.xml` and replace the `<colors> ... </colors>` section with the following:

Listing 9.3 : Solarized Color for Console2

```
<colors>
  <color id="0" r="7" g="54" b="66"/>         <!-- black -->
  <color id="1" r="38" g="139" b="210"/>      <!-- blue -->
  <color id="2" r="133" g="153" b="0"/>       <!-- green -->
  <color id="3" r="42" g="161" b="152"/>      <!-- cyan -->
  <color id="4" r="220" g="50" b="47"/>       <!-- red -->
  <color id="5" r="211" g="54" b="130"/>      <!-- magenta -->
  <color id="6" r="181" g="137" b="0"/>       <!-- yellow/brown -->
  <color id="7" r="238" g="232" b="213"/>     <!-- white -->
  <color id="8" r="0" g="43" b="54"/>         <!-- brblack -->
  <color id="9" r="131" g="148" b="150"/>     <!-- brblue -->
  <color id="10" r="88" g="110" b="117"/>     <!-- brgreen -->
  <color id="11" r="147" g="161" b="161"/>    <!-- brcyan -->
  <color id="12" r="203" g="75" b="22"/>      <!-- brred -->
  <color id="13" r="108" g="113" b="196"/>    <!-- brmagenta/violet -->
  <color id="14" r="101" g="123" b="131"/>    <!-- bryellow -->
  <color id="15" r="253" g="246" b="227"/>    <!-- brwhite  -->
</colors>
```

Changing ZSH Theme Oh-My-ZSH comes with several already prepared themes. To use one of these just change `ZSH_THEME="THEME_NAME"` in `~/.zshrc`, **bureau** is a good looking theme on Cygwin. Some of the themes require installing some patched fonts like patched **mensch-powerline** or all the other Power Line fonts (specially for themes like agnoster).

[1] like `compdef: unknown command or service: git`

Get rid of long User@Host string Inorder to get rid of long user@host string in some of the themes you can add the following lines to `~/.zhrc` :

Listing 9.4 : Removing long User@Host string

```
DEFAULT_USER=`whoami`
user=`whoami`
HOST="local"
```

After changing the hostname you might encounter the `compdef: unknown command or service: git` error again which could be resolved as discussed in (9.3.1). To see all the available used environmental variables use the command `typeset` .

How to copy current path and use it in another terminal ?
I couldn't find a way to do it automatically. The closest thing is using xclip equivalent which is getclip/putclip. To install them use `apt-cyg install cygutils-extra` command.

Listing 9.5 : Using aliases to push/pop last directory

```
alias gp="pwd|putclip"
alias up="cd `getclip`"
alias getpath="pwd|putclip"
alias usepath="cd `getclip`"
```

A better way of doing this is to use a localfile to write last changed directory result.

Listing 9.6 : Using aliases and a local file to push/pop last directory

```
cdcopy() { cd "$@" ; pwd>~/.lastfolder}
cc() { pwd>~/.lastfolder}
usecopiedpath() {lastfolder=$(cat ~/.lastfolder) && cd $lastfolder }
usecopiedpath
alias cd=cdcopy
alias uc=usecopiedpath
```

Appendices

A

chrome extensions

A.1 Supporting LiveEdit with gulp

In the extension folder (app) create a gulpfile.js as below:

Listing A.1: gulpfile.js

```
/**
* Created by Reza on 1/31/2015.
*/
var gulp = require('gulp');
var tinylr = require('tiny-lr');
var EXPRESS_PORT = 4000;
var EXPRESS_ROOT = __dirname;

function startExpress() {

  var express = require('express');
  var app = express();
  app.use(require('connect-livereload')());
  app.use(express.static(EXPRESS_ROOT));
  app.listen(EXPRESS_PORT);
}

gulp.task('dev', function () {
  startExpress();
  var lr = tinylr();
  var port = 35729;
  lr.listen(port);
  gulp.watch(['**/*.{js,css,html,json}'], function (evt) {
    console.log('Event path: ' + evt.path);
    lr.changed({
      body: {
        files: [evt.path]
      }
    });
  });
});
```

A.2 Support MouseHover with delay

Listing A.2 : Anchor Mouse Hover handling with delay

```
var popUpTimeout = null;
function doMouseIn(anchor, href) {
  popUpTimeout = setTimeout(function () {
    console.log(href);
    anchor.css("background-color", "blue");
  }, 1000);
}
function doMouseOut() {
  clearTimeout(popUpTimeout);
}
```

A

Android Upgrade

A.1 PreRequirements

There are certain considerations before upgrade for different operating systems:

A.1.1 Linux

Running commands as admin

In linux both adb and fastboot commands should run as administrator (sudo). But they are not accessible for root user by default, to make them accessible use:

```
ln -s /home/existme/Android/Sdk/platform-tools/fastboot /usr/local/sbin/fastboot
ln -s /home/existme/Android/Sdk/platform-tools/adb /usr/local/sbin/adb
```

A.1.2 Windows

N/A

A.1.3 Mac

No need to do anything.

A.2 Upgrade process

When a new upgrade comes, there are two ways to upgrade the android device:

A.2.1 Factory Reset/Flashing Stock

Factory reset is the easiest option. First download latest factory image for your device from this link.

- **Are you connected? :**
  First, make sure that your computer is communicating correctly with your Nexus phone or tablet. use the command: `adb devices`
- **Bootloader:** Boot into bootloader using: `adb reboot bootloader`
- **Unlock OEM if needed:**
  If your device is not already unlocked it's essential that you unlock it before proceeding to the next steps. Unlock your device using: `fastboot oem unlock`

- **FlashAll:**
  Execute the flash-all script. This script installs the necessary bootloader, baseband firmware(s), and operating system.
  `./flash-all`

A.2.2 OTA update

First download the OTA update, you should search for the id and keyword OTA. for example search for "LMY47I OTA link"

- **Are you connected? :**
  First, make sure that your computer is communicating correctly with your Nexus phone or tablet. use the command: `adb devices`

- **Bootloader:** Boot into bootloader using: `adb reboot bootloader`

- **Goto Recovery mode:**
  Use the **volume down** button twice until you have scrolled to **Recovery mode**, and press the power button to select it

- **Don't freak:** An image of an Android with a red exclamation mark over it will appear (that's OK)

- ****Get out of this mode:**
  Hold down the **power button** and press the **volume up button**, and you will be in recovery mode

- **Go to receiving adb mode:**
  Use the volume down button to highlight **apply update from ADB** and press the power button to select it

- **Flash the actual system image:** Flash the actual system image using:
  `sudo adb sideload [OTA file].zip`

A.2.3 Do you have problem with *adb sideload*?

For linux machines I've encountered a problem in which `adb sideload` wasn't able to upload the file. When issuing " `adb sideload devices` ", I just got `????????????? no permissions` .
This problem is due to wrong adb-server state. If you restart the server the problem would be fixed:

```
sudo su
adb kill-server
adb start-server
exit
```

For Rooted systems: Usually if you are rooted, you can't do it just like the above procedure and last step would fail. That's because system files are modified and the updater is going to verify system files. The easiest way to overcome this problem is:

- **Download latest stock image:**
  First, download the exact stock image of your OTA image from here. Note that these are two different files. The stock image is bigger than the OTA image.

- **Extract needed files from stock image**
  Extract big zip file from the downloaded `.tgz` file.
  Extract system.img boot.img and recovery.img from the extracted zip file.

- **Flash system files**
  Flash the extracted files to your device. Note that your data wouldn't be lost in this way.
  ```
  sudo fastboot flash boot boot.img
  sudo fastboot flash recovery recovery.img
  sudo fastboot flash system system.img
  ```
- **Retry to update OTA**
  Now retry to update OTA using the above procedure.

Note! There is a `flash_all.sh` script which can be used in-case you weren't able to unroot the device and all the above methods failed. However, it will completely ***wipe*** the device.

A.3 Disabling the encryption

In some devices it's not possible to remove encryption easily. To disable it and improving performance you need to use a modified boot loader. The only option that I am currently aware of, is a windows-base utility called "Nexus Root Toolkit" which can do the job but this process needs wiping the device. However, this utility is extremely powerful since it can almost automatically do everything including the above steps. The only downside is that it is only available for Windows.

A.4 Rooting

The fastest way to root nexus devices:

- **Get the latest TWRP:** Get the latest TWRP image for your device for example from here.
- **Get the latest SUPERSU:** Get the latest SuperSU **zip** for your device from chainfire.
- **Goto bootloader:** Boot into bootloader using: `adb reboot bootloader`
- **Boot using TWRP** Boot the device with TWRP image downloaded using:
  `sudo fastboot boot [twrp-image-file].img`
- **Sideload SUPERSU** In TWRP screen start sideload process, then sideload downloaded supersu zip file using: `sudo adb sideload [SuperSU-file].zip`
- **Reboot** Reboot your device, install SUPERSU application and RootChecker to verify your installation.