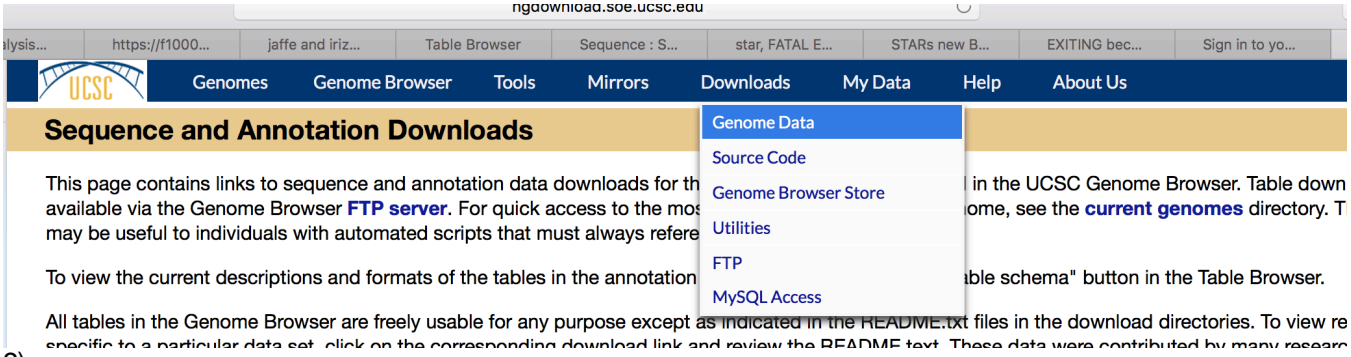
RNAseq workshop Day2 (organized)

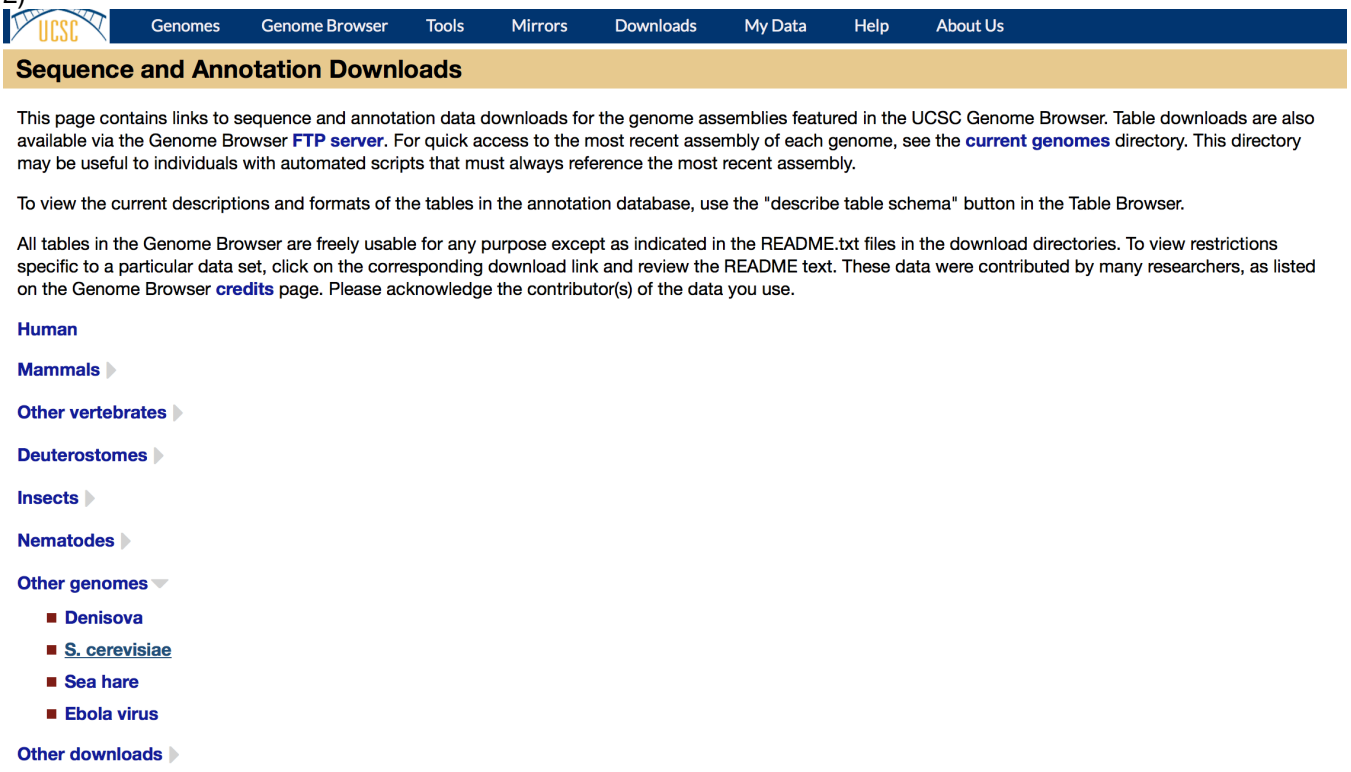
1. Experimental design
2. Alignment: 1) find the reference genome; 2) find the annotation file; 3) run STAR for alignment; 4) use samtools to check the generated SAM files after alignment; 5) use IGV to visualize the alignment results

Part 1. Find the reference genome:
the reference genome doesn't change that often, maybe once several years; sometimes don't use the reference genome just published several months ago, because the annotation file might haven't updated yet.

1. Download the reference genome:
Go to UCSD GenomeBrowser and do it as the following instructions:
Go to "Downloads", choose "Genome Data", choose "S. cerevisiae" genome, go to "sacCer3" "full data set", scroll down, download ".2bit" file with the following code:
download the .2bit file with the link
wget http://hgdownload.soe.ucsc.edu/goldenPath/sacCer3/bigZips/sacCer3.2bit
turn compressed 2bit format into FASTA format (.fa)
UCSC-tools/twoBitToFa sacCer3.2bit sacCer3.fa
(for the workshop, these files are already downloaded into "`mat/referenceGenomes/S_cerevisiae/`"
1)



2)



3)

Yeast (S. cerevisiae) genome

Apr. 2011 (SacCer_Apr2011/sacCer3)

- Full data set
- Data set by chromosome
- Annotation database
- LiftOver files
- Multiple alignments

4)

Name	Last modified	Size	Description
Parent Directory		-	
chromAgp.tar.gz	02-Sep-2011 14:19	660	
chromFa.tar.gz	02-Sep-2011 14:19	3.6M	
chromFaMasked.tar.gz	02-Sep-2011 14:20	3.6M	
chromTrf.tar.gz	02-Sep-2011 14:20	20K	
est.fa.gz	17-Aug-2017 08:46	6.2M	
est.fa.gz.md5	17-Aug-2017 08:46	44	
md5sum.txt	02-Sep-2011 14:20	251	
mrna.fa.gz	17-Aug-2017 08:40	111K	
mrna.fa.gz.md5	17-Aug-2017 08:40	45	
sacCer3.2bit	24-Aug-2011 11:55	2.9M	
sacCer3.beta.tables	05-Oct-2011 16:10	1.5K	
sacCer3.chrom.sizes	24-Aug-2011 11:55	229	
sacCer3.tables	05-Oct-2011 16:10	2.9K	
upstream1000.fa.gz	17-Aug-2017 08:47	11K	
upstream1000.fa.gz.md5	17-Aug-2017 08:47	53	
upstream2000.fa.gz	17-Aug-2017 08:47	21K	
upstream2000.fa.gz.md5	17-Aug-2017 08:47	53	
upstream5000.fa.gz	17-Aug-2017 08:47	52K	
upstream5000.fa.gz.md5	17-Aug-2017 08:47	53	
xenoRefMrna.fa.gz	17-Aug-2017 08:47	303M	
xenoRefMrna.fa.gz.md5	17-Aug-2017 08:47	52	

2. to check the downloaded reference genome:

[ngscls33@scu-node02 ~]$ egrep "<chr" mat/referenceGenomes/S_cerevisiae/sacCer3.fa

[ngscls33@scu-node02 ~]$ egrep ">chr" mat/referenceGenomes/S_cerevisiae/sacCer3.fa

```
[ngscls33@scu-node02 ~]$ egrep "<chr" mat/referenceGenomes/S_cerevisiae/sacCer3.
fa
[ngscls33@scu-node02 ~]$ egrep ">chr" mat/referenceGenomes/S_cerevisiae/sacCer3.
fa
>chrI
>chrII
>chrIII
>chrIV
>chrIX
>chrV
>chrVI
>chrVII
>chrVIII
>chrX
>chrXI
>chrXII
>chrXIII
>chrXIV
>chrXV
>chrXVI
>chrM
```

[ngscls33@scu-node02 ~]$ head mat/referenceGenomes/S_cerevisiae/sacCer3.fa

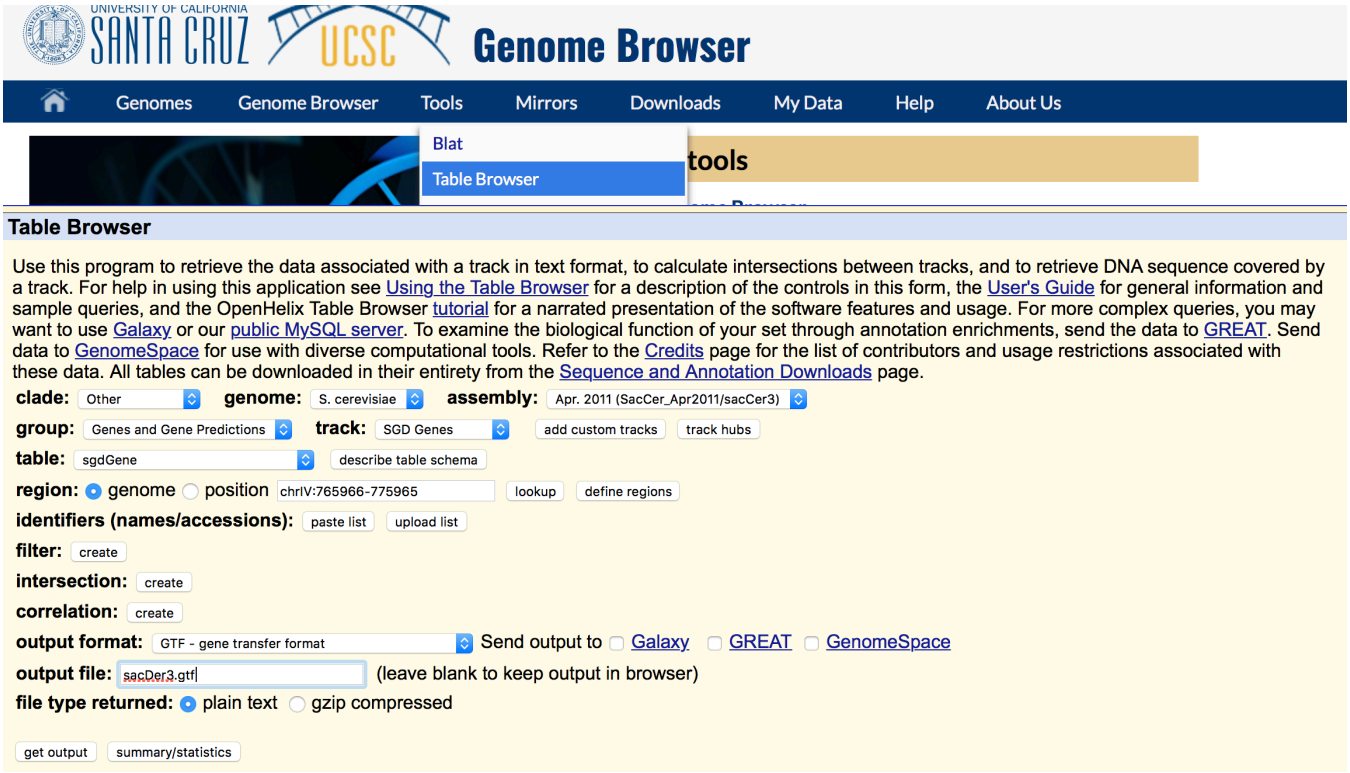
```
[ngscls33@scu-node02 ~]$ head mat/referenceGenomes/S_cerevisiae/sacCer3.fa
>chrI
CCACACCACACCCACACACCCACACACCACACCACACACCACACCACACC
CACACACACACATCCTAACACTACCCTAACACAGCCCTAATCTAACCCTG
GCCAACCTGTCTCTCAACTTACCCTCCATTACCCTGCCTCCACTCGTTAC
CCTGTCCCATTCAACCATACCACTCCGAACCACCATCCATCCCTCTACTT
ACTACCACTCACCCACCGTTACCCTCCAATTACCCATATCCAACCCACTG
CCACTTACCCTACCATTACCCTACCATCCACCATGACCTACTCACCATAC
TGTTCTTCTACCCACCATATTGAAACGCTAACAAATGATCGTAAATAACA
CACACGTGCTTACCCTACCACTTTATACCACCACCACATGCCATACTCAC
CCTCACTTGTATACTGATTTTACGTACGCACACGGATGCTACAGTATATA
```

##

Part II. Find the annotation file
Annotation is dynamic and updated for several times a year. It includes the sequence, coordinates, types of elements information, based on published data and predicted information.
Different sources of annotation files could have various results. Commonly used ones include: 1) RefSeq (from NCBI): the most conservative one. 2) UCSC Known Genes (from UCSC genome browser) 3) Ensembl/Gencode: the most sensitive one with the largest number of annotated genes

Choose the annotation data base depends on the project. Use the one that is generally used by people in the field or lab is a good choice.

1. Download the annotation from UCSC genome browser (the annotation file is in GTF format, compatilble with downstream STAR).
Go to “Table Browser” under “Tools”, then choose proper parameters. Type “sacDer3.gtf” in the output filie



2. Properly format the downloaded sacDer3.gtf files

> **!** GTF files downloaded from the UCSC Table Browser have the same entries for `gene_id` and `transcript_id`. This can lead to problems with downstream analysis tools that expect exons of different isoforms to have the same `gene_id`, but different `transcript_ids`.

Here's a way to get a properly formatted `GTF` file (i.e., with different entries for `gene_name` and `transcript_id`) of RefSeq genes using the UCSC tool `genePredToGtf`:

```
# first, download a table for "Genes and Gene Predictions" from the UCSC Table Browser indicating as the output format: "all fields from selected table"
# NOTE: this may not work for all GTF files downloaded from UCSC! genePredToGtf is very finicky and every organism's annotation may have been generated and deposited by a different person)
$ head -n1 allfields_hg19.txt
bin	name	chrom	strand	txStart	txEnd	cdsStart	cdsEnd	exonCount	exonStarts	exonEnds	score	name2	cdsStartStat	cdsEndStatexonFrames
# remove first column and first line, feed that into genePredToGtf
$ cut -f 2- allfields_hg19.txt | sed '1d' | \
  genePredToGtf file stdin hg19_RefSeq.gtf
$ head -n1 hg19_RefSeq.gtf
chr1	stdin	exon	66999639	67000051	.	+	.	gene_id "SGIP1"; transcript_id "NM_032291"; exon_number "1"; exon_id "NM_032291.1"; gene_name "SGIP1";
```

In this workshop, the processed GTF file is stored in as "sacCer3.gtf" in "`referenceGenomes/S_cerevisiae/`"

##

Part 3. Run the STAR alignment

1. Do some extra steps to simplify the following steps:

#1.1 Define a variable to have easy access to the directory that stores the "reference genome" and "annotation"

```
[ngscls33@scu-node02 ~]$ REF_DIR=~/mat/referenceGenomes/S_cerevisiae
[ngscls33@scu-node02 ~]$ ls $REF_DIR
[ngscls33@scu-node02 ~]$ REF_DIR=~/mat/referenceGenomes/S_cerevisiae
[ngscls33@scu-node02 ~]$ ls $REF_DIR
README        sacCer3.bed  sacCer3.gtf                                STARindex
sacCer3.2bit  sacCer3.fa   Saccharomyces_cerevisiae.R64-1-1.81.gtf

#1.2 make a new directory to store the aligned results:
[ngscls33@scu-node02 ~]$ mkdir alignment_STAR

#1.3 Define a variable to have easy access to the STAR program
[ngscls33@scu-node02 ~]$ runSTAR=~/mat/software/STAR-2.5.3a/bin/Linux_x86_64_static/STAR
[ngscls33@scu-node02 ~]$ $runSTAR --help
# –help to confirm $runSTAR will work properly

#1.4 store all the fastq.gz files into a variable named “FILES”. Clean up “FILES” to remove
unnecessary spaces as the input for $runSTAR

# the files are the downloaded fastq.gz files (see day1 for instruction how to download them).
The files used today are stored in ~/mat/rawReads_yeast_Gierlinski/WT_1/

[ngscls33@scu-node02 ~]$ FILES=`ls -m ~/mat/rawReads_yeast_Gierlinski/WT_1/*fastq.gz | sed 's/
//g'`
$ echo $FILES

[ngscls33@scu-node02 ~]$ FILES=`ls -m ~/mat/rawReads_yeast_Gierlinski/WT_1/*fast
q.gz | sed 's/ //g'`
[ngscls33@scu-node02 ~]$ echo $FILES
/home/ngscls33/mat/rawReads_yeast_Gierlinski/WT_1/ERR458493.fastq.gz, /home/ngsc
ls33/mat/rawReads_yeast_Gierlinski/WT_1/ERR458494.fastq.gz, /home/ngscls33/mat/r
awReads_yeast_Gierlinski/WT_1/ERR458495.fastq.gz, /home/ngscls33/mat/rawReads_ye
ast_Gierlinski/WT_1/ERR458496.fastq.gz, /home/ngscls33/mat/rawReads_yeast_Gierli
nski/WT_1/ERR458497.fastq.gz, /home/ngscls33/mat/rawReads_yeast_Gierlinski/WT_1/
ERR458498.fastq.gz, /home/ngscls33/mat/rawReads_yeast_Gierlinski/WT_1/ERR458499.
fastq.gz
# here $FILES still have unnecessary spaces


# to remove additioanl spaces:
[ngscls33@scu-node02 ~]$ FILES=`echo $FILES | sed 's/ //g'`

[ngscls33@scu-node02 ~]$ FILES=`echo $FILES | sed 's/ //g'`
[ngscls33@scu-node02 ~]$ echo $FILES
/home/ngscls33/mat/rawReads_yeast_Gierlinski/WT_1/ERR458493.fastq.gz,/home/ngscl
s33/mat/rawReads_yeast_Gierlinski/WT_1/ERR458494.fastq.gz,/home/ngscls33/mat/raw
Reads_yeast_Gierlinski/WT_1/ERR458495.fastq.gz,/home/ngscls33/mat/rawReads_yeast
_Gierlinski/WT_1/ERR458496.fastq.gz,/home/ngscls33/mat/rawReads_yeast_Gierlinski
/WT_1/ERR458497.fastq.gz,/home/ngscls33/mat/rawReads_yeast_Gierlinski/WT_1/ERR45
8498.fastq.gz,/home/ngscls33/mat/rawReads_yeast_Gierlinski/WT_1/ERR458499.fastq.
gz
```

2. Generate genome index

Generate genome index This step has to be done only once per genome type (and alignment

Generate genome index This step has to be done only once per genome type (and alignment program). The index files will comprise the genome sequence, suffix arrays, chromosome names and lengths, splice junctions coordinates, and information about the genes. Therefore, the main input for this step encompasses the reference genome and an annotation file.

```
# create a directory to store the index in
$ REF_DIR=mat/referenceGenomes/S_cerevisiae
$ mkdir ~/STARindex

# set a variable for STAR access
$ runSTAR=mat/software/STAR-2.5.3a/bin/Linux_x86_64_static/STAR

# Run STAR in "genomeGenerate" mode
$ ${runSTAR} --runMode genomeGenerate \
    --genomeDir ~/STARindex \ # index will be stored there
    --genomeFastaFiles ${REF_DIR}/sacCer3.fa \ # reference genome sequence
    --sjdbGTFfile ${REF_DIR}/sacCer3.gtf \ # annotation file
    --sjdbOverhang 49 # should be read length minus 1 ; length of the
       genomic sequence around the annotated junction to be used for the
       splice junctions database
    --runThreadN 1 \ # can be used to define more processors
```

```
[ngscls33@scu-node02 ~]$ ls ~/mat/referenceGenomes/S_cerevisiae/
README         sacCer3.bed  sacCer3.gtf                          STARindex
sacCer3.2bit   sacCer3.fa   Saccharomyces_cerevisiae.R64-1-1.81.gtf
```

3. Do alignment. Run STAR with the reference genome, annotation and the fastq.gz files(sequencing results)

```
[ngscls33@scu-node02 ~]$ $runSTAR --genomeDir
/home/ngscls33/mat/referenceGenomes/S_cerevisiae/STARindex/ \
--readFilesIn $FILES \  #read in the fastq.gz files
--readFilesCommand zcat \  #unzip the fastq.gz files
--outFileNamePrefix alignment_STAR/WT_1_ \  #directory to store the alinged results
--outFilterMultimapNmax 1 \  #only reads with 1 match in the reference will be returned as
aligned
--outReadsUnmapped Fastx \  #will generate an extra output file with the ualigned reads
--outSAMtype BAM SortedByCoordinate \  #format of output files
--twopassMode Basic \   #STAR will extract novel junctions????????
--runThreadN 1  #??????/
```

#this is how the progress looks like when running alignment:

```
Aug 18 13:21:20 ..... started STAR run
Aug 18 13:21:20 ..... loading genome
Aug 18 13:21:23 ..... started 1st pass mapping
Aug 18 13:24:26 ..... finished 1st pass mapping
Aug 18 13:24:26 ..... inserting junctions into the genome indices
Aug 18 13:24:33 ..... started mapping
Aug 18 13:27:45 ..... started sorting BAM
[W::bam_hdr_read] EOF marker is absent. The input is probably truncated.
[bam_cat] WARNING: Unexpected block structure in file 'alignment_STAR/WT_1__STAR
tmp//BAMsort//b3'. Possible output corruption.
Aug 18 13:28:07 ..... finished successfully
```

#After running, these are the newly generated results:

```
[ngscls33@scu-node02 ~]$ ls alignment_STAR/
WT_1_Aligned.sortedByCoord.out.bam  WT_1Log.progress.out
WT_1Aligned.sortedByCoord.out.bam   WT_1_SJ.out.tab
WT_1_Log.final.out                  WT_1__STARgenome
WT_1_Log.out                        WT_1__STARpass1
WT_1Log.out                         WT_1_STARtmp
WT_1_Log.progress.out               WT_1_Unmapped.out.mate1
```

4. BAM File indexing

BAM file indexing Most downstream applications will require a `.BAM.BAI` file together with every `BAM` file to quickly access the `BAM` files without having to load them into memory. To obtain these index files, simply run the `samtools index` command for each `BAM` file once the mapping is finished.

```
# export samtools path (for convenience)
$ export PATH=/zenodotus/abc/store/courses/2017_rnaseq/software/samtools
    -1.5/bin:$PATH

# index the BAM file
$ samtools index alignment_STAR/WT_1_Aligned.sortedByCoord.out.bam
```

```
################################################################################

Part 4. Use samtools to view the aligned reads
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools –help

# Check the first several lines:
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view
mat/results_alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam | head

ERR458493.552967        16      chrI    140     255     12M61232N37M2S  *       C
CACTCGTTCACCAGGGCCGGCGGGCTGATCACTTTATCGTGCATCTTGGC      BB?HHJJIGHHJIGIIJJIJGIJI
JJIIIGHBJJJJJJHHHHFFDDDA1+B     NH:i:1  HI:i:1  AS:i:41 nM:i:2
ERR458493.889005        256     chrI    1850    0       51M     *       0       G
AGTTGGCTGGCTTTAATCTGCTGGAGTACCATGGAACACCGGTGATCATT      :+=4=22AC;CFFIIIEGICHGII
IBIGHHHFAECGGHIBFA@?@BFC>FB     NH:i:6  HI:i:2  AS:i:48 nM:i:1
ERR458493.784623        16      chrI    1934    0       51M     *       0       A
TGGTGGTGAAGTCACCGTAGTTGAAAACGGCTTCAGCAACTTCGACTGGG      IIIIIIIIGIGIIHFIEGF@IGII
IIIHIIIIIIIIIIHHHHHDDDD?=;@     NH:i:7  HI:i:1  AS:i:50 nM:i:0
ERR458493.303774        16      chrI    1944    0       51M     *       0       A
GTCACCGTAGTTGAAAACGGCTTCAGCAACTTCGACTGGGTAGGTTTCAG      FBCB8IIJJJIIJIIJHGHIJIII
IHDGJJIJJIIHJIHHHHHFFFFFCC@     NH:i:7  HI:i:1  AS:i:50 nM:i:0
ERR458493.748218        272     chrI    1944    0       51M     *       0       A
```

check the first result:

```
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view
mat/results_alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam | head -n1
```

```
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view mat/results_a
lignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam | head -n1
ERR458493.552967        16      chrI    140     255     12M61232N37M2S  *       C
CACTCGTTCACCAGGGCCGGCGGGCTGATCACTTTATCGTGCATCTTGGC      BB?HHJJIGHHJIGIIJJIJGIJI
JJIIIGHBJJJJJJHHHHFFDDDA1+B     NH:i:1  HI:i:1  AS:i:41 nM:i:2
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -H mat/result
```

#return the header and alignment section:

```
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -H
mat/results_alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -H mat/results_
alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam
@HD     VN:1.4  SO:coordinate
@SQ     SN:chrI LN:230218
@SQ     SN:chrII        LN:813184
@SQ     SN:chrIII       LN:316620
@SQ     SN:chrIV        LN:1531933
@SQ     SN:chrIX        LN:439888
@SQ     SN:chrV LN:576874
@SQ     SN:chrVI        LN:270161
@SQ     SN:chrVII       LN:1090940
@SQ     SN:chrVIII      LN:562643
@SQ     SN:chrX LN:745751
@SQ     SN:chrXI        LN:666816
@SQ     SN:chrXII       LN:1078177
@SQ     SN:chrXIII      LN:924431
@SQ     SN:chrXIV       LN:784333
@SQ     SN:chrXV        LN:1091291
@SQ     SN:chrXVI       LN:948066
@SQ     SN:chrM LN:85779
@PG     ID:STAR PN:STAR VN:STAR_2.5.3a  CL:./software/STAR-2.5.3a/bin/Linux_x86_64
_static/STAR   --runThreadN 8   --genomeDir /zenodotus/abc/store/courses/2017_rnas
eq/referenceGenomes/S_cerevisiae/STARindex/   --readFilesIn /zenodotus/abc/store/c
ourses/2016_rnaseq/rawReads_yeast_Gierlinski/WT_1/ERR458493.fastq.gz      --readFi
lesCommand zcat      --outFileNamePrefix aln_tmp/WT_1_   --outReadsUnmapped Fastx
  --outSAMtype BAM   SortedByCoordinate      --twopassMode Basic
@CO     user command line: ./software/STAR-2.5.3a/bin/Linux_x86_64_static/STAR --g
enomeDir /zenodotus/abc/store/courses/2017_rnaseq/referenceGenomes/S_cerevisiae/ST
ARindex/ --readFilesIn /zenodotus/abc/store/courses/2016_rnaseq/rawReads_yeast_Gie
rlinski/WT_1/ERR458493.fastq.gz --readFilesCommand zcat --outFileNamePrefix aln_tm
p/WT_1_ --runThreadN 8 --twopassMode Basic --outReadsUnmapped Fastx --outSAMtype B
AM SortedByCoordinate
```

#return the header and include only the mapped reads, show the folders

```
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -h -f 4
mat/results_alignment/WT_1/WT_1_
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -h -f 4 mat/res
ults_alignment/WT_1/WT_1_
WT_1_Aligned.out.sam                WT_1_SJ.out.tab
WT_1_Aligned.sortedByCoord.out.bam  WT_1__STARgenome/
WT_1_Log.final.out                  WT_1__STARpass1/
WT_1_Log.out                        WT_1_Unmapped.out.mate1
WT_1_Log.progress.out
```

#return the header and include only the mapped reads, show the chromosome information

```
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -h -f 4
mat/results_alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam | head
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -h -f 4 mat/res
```

```
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -h -F 4 mat/res
ults_alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam | head
@HD     VN:1.4  SO:coordinate
@SQ     SN:chrI LN:230218
@SQ     SN:chrII        LN:813184
@SQ     SN:chrIII       LN:316620
@SQ     SN:chrIV        LN:1531933
@SQ     SN:chrIX        LN:439888
@SQ     SN:chrV LN:576874
@SQ     SN:chrVI        LN:270161
@SQ     SN:chrVII       LN:1090940
@SQ     SN:chrVIII      LN:562643

#do counting for mapped reads:
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -h -F 4
mat/results_alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam | wc -l
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -h -F 4 mat/res
ults_alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam | wc -l
1182855

#do counting for unmapped reads:
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -h -f 4
mat/results_alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam | wc -l
[ngscls33@scu-node02 ~]$ ~/mat/software/samtools-1.5/samtools view -h -f 4 mat/res
ults_alignment/WT_1/WT_1_Aligned.sortedByCoord.out.bam | wc -l
20

#########################################################################

Part V Using IGV to visualize mapped reads

# 1. Download the files with wget on Mac:
mac184457:downloads fanyingtang$ wget
"http://chagall.med.cornell.edu/RNASEQcourse/WT_1_Aligned.sortedByCoord.out.bam”
mac184457:downloads fanyingtang$ wget
"http://chagall.med.cornell.edu/RNASEQcourse/WT_1_Aligned.sortedByCoord.out.bam.bai"
```