```
meta.file <- "PRJNA835826-meta.csv"
meta.info <- read.csv(meta.file, header = TRUE)
print(meta.info)

# Download the SRA normalized format files
for(run in meta.info$Run) {
  cmd <- paste("prefetch", run, "--verbose")
  cat(cmd, "\n")
  system(cmd)
}

# Retrieve the fastq files
for(run in meta.info$Run) {
  fq.files <- paste0(run, "_", 1:2, ".fastq")
  if(!all(file.exists(fq.files))) {
    cmd <- paste("fasterq-dump --threads 12 --verbose", run)
    cat(cmd, "\n")
    system(cmd)

  } else {
    cat(fq.files, "already present. Skipped.\n")
  }

}
```

```
STAR --runThreadN 8 \
    --runMode genomeGenerate \
    --genomeDir /home/kraken2/2023-489-509/zcui/index/genome_index_directory \
    --genomeFastaFiles /home/kraken2/2023-489-509/zcui/index/genome.fa \
```

```
    --sjdbGTFfile /home/kraken2/2023-489-509/zcui/index/genes.gtf \
    --sjdbOverhang 99



cd /home/kraken2/2023-489-509/zcui/project2
for i in $(seq 13 24); do
  STAR --runThreadN 8 --genomeDir /home/kraken2/2023-489-
509/zcui/index/genome_index_directory --readFilesIn SRR191232${i}_1.fastq
SRR191232${i}_2.fastq --outFileNamePrefix SRR191232${i}_ --outSAMtype BAM
SortedByCoordinate




# Load the parallel package

library(parallel)


# Set paths

gtf_path <- "/home/kraken2/2023-489-
509/zcui/index/gencode.vM33.primary_assembly.basic.annotation.gtf"

output_dir <- "/home/kraken2/2023-489-509/zcui/project2/count/"


# Function to run htseq-count

run_htseq <- function(bam_file) {

  base_name <- sub(".bam", "", basename(bam_file))

  output_file <- paste0(output_dir, base_name, "_counts.txt")

  cmd <- sprintf("htseq-count -f bam -s no %s %s > %s", bam_file, gtf_path, output_file)

  system(cmd)

  return(sprintf("Processed %s, output saved to %s", bam_file, output_file))

}


# List of bam files

bam_files <- c("SRR19123213_Aligned.sortedByCoord.out.bam",
```

```
        "SRR19123214_Aligned.sortedByCoord.out.bam",
        "SRR19123215_Aligned.sortedByCoord.out.bam",
        "SRR19123216_Aligned.sortedByCoord.out.bam",
        "SRR19123217_Aligned.sortedByCoord.out.bam",
        "SRR19123218_Aligned.sortedByCoord.out.bam",
        "SRR19123219_Aligned.sortedByCoord.out.bam",
        "SRR19123220_Aligned.sortedByCoord.out.bam",
        "SRR19123221_Aligned.sortedByCoord.out.bam",
        "SRR19123222_Aligned.sortedByCoord.out.bam",
        "SRR19123223_Aligned.sortedByCoord.out.bam",
        "SRR19123224_Aligned.sortedByCoord.out.bam")

# Number of cores for parallel processing
num_cores <- detectCores() - 1

# Use parallel processing to run htseq-count on each bam file
results <- mclapply(bam_files, run_htseq, mc.cores = num_cores)

# Print results
for (res in results) {
  cat(res, "\n")
}
```