

# Snakemake

Elementi di bioinformatica

# Basic Syntax

```
rule RULE_NAME:
    input:
        "INPUT_FILE"
    output:
        "OUTPUT_FILE"
    shell:
        '''
            Command to execute using input and output
        '''
```

# Basic Syntax

```
rule cat:
    input:
        "test.fa"
    output:
        "catted.fa"
    shell:
        '''
            cat {input} > {output}
        '''
```

# Example rules

```
rule cat:
    input:
        f_in="test.fa"
    output:
        f_out="catted.fa"
    shell:
        '''
            cat {input.f_in} > {output.f_out}
        '''
```

# Example rules

```
rule cat:
    input:
        f_in1="fa1.fa",
        f_in2="fa2.fa",
    output:
        f_out="catted.fa"
    shell:
        '''
            cat {input.f_in1} {input.f_in2} > {output.f_out}
        '''
```

# Example rules

```
rule wc:  
  input:  
    "catted.fa"  
  output:  
    count=wc.txt  
  shell:  
    '''  
        wc {input} > {output.count}  
    '''
```

# Example rules

```
rule grep:
  input:
    catted="catted.fa",
  output:
    s="seqs.txt",
    a="a.txt",
    c="c.txt",
  shell:
    '''
    grep "^>" {input.catted} -c > {output.g}
    grep "[Aa]" {input.catted} -c > {output.a}
    grep "[Cc]" {input.catted} -c > {output.c}
    '''
```

# How to run the workflow

Save the rules in one file called “Snakefile”

```
snakemake
```

Save to a file called “test.smk”

```
snakemake -s test.smk
```



# More realistic example

```
rule index_fa:
    input:
        "hs37d5.fa"
    output:
        "hs37d5.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

```
rule get_region:
    input:
        fa="hs37d5.fa",
        fai="hs37d5.fa.fai",
    output:
        "sub.fa"
    params:
        reg="21:34407410-40062514",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```

# More realistic example

```
rule index_fa:
    input:
        "hs37d5.fa"
    output:
        "hs37d5.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

```
rule get_region:
    input:
        fa="hs37d5.fa",
        fai="hs37d5.fa.fai",
    output:
        "sub.fa"
    params:
        reg="21:34407410-40062514",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```

# More realistic example

```
rule index_fa:
    input:
        "hs37d5.fa"
    output:
        "hs37d5.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

```
rule get_region:
    input:
        fa="hs37d5.fa",
        fai="hs37d5.fa.fai",
    output:
        "sub.fa"
    params:
        reg="21:34407410-40062514",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```

snakemake -s test.smk -c1

# More realistic example

```
rule index_fa:
    input:
        "hs37d5.fa"
    output:
        "hs37d5.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

```
rule get_region:
    input:
        fa="hs37d5.fa",
        fai="hs37d5.fa.fai",
    output:
        "sub.fa"
    params:
        reg="21:34407410-40062514",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```

snakemake -s test.smk -c1 -f get\_region

# Rle *run* at the beginning

```
rule run:  
    input:  
        "sub.fa"
```

```
snakemake -s test.smk -c1
```

# DAG

snakemake -s test.smk --dag

```
digraph snakemake_dag {  
    graph[bgcolor=white, margin=0];  
    node[shape=box, style=rounded, fontname=sans, fontsize=10, penwidth=2];  
    edge[penwidth=2, color=grey];  
    0[label = "run", color = "0.22 0.6 0.85", style="rounded,dashed"];  
    1[label = "get_region", color = "0.44 0.6 0.85", style="rounded,dashed"];  
    2[label = "index_fa", color = "0.00 0.6 0.85", style="rounded,dashed"];  
    1 -> 0  
    2 -> 1  
}
```



# Expand

```
rule index_fa:
    input:
        "{sample}.fa"
    output:
        "{sample}.fa.fai"
    shell:
        '''
        samtools faidx {input}
        '''
```

# Expand

```
rule get_region:
    input:
        fa="{sample}.fa",
        fai="{sample}.fa.fai",
    output:
        "{sample}.sub.fa"
    params:
        reg="21:100-200",
    shell:
        '''
        samtools faidx {input.fa} {params.reg} > {output}
        '''
```



# Expand

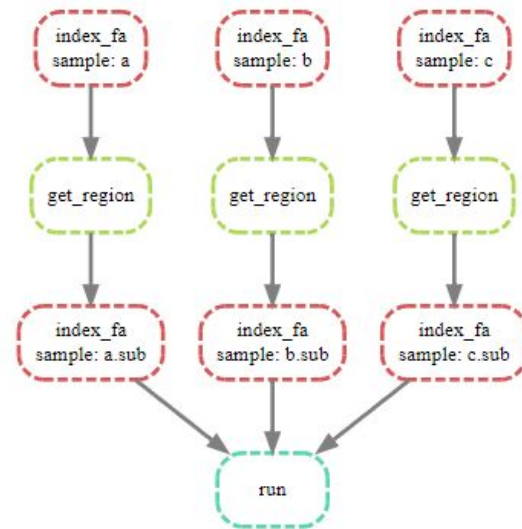
```
rule run:
    input:
        expand(
            "{sample}.sub.fa.fai",
            sample=["a", "b", "c"]
        )
```

# Expand

```
rule run:
    input:
        expand(
            "{sample}.sub.fa.fai",
            sample=["a", "b", "c"]
        )
```

# Expand

```
rule run:
    input:
        expand(
            "{sample}.sub.fa.fai",
            sample=["a", "b", "c"]
        )
```



# The expand function - 2

```
rule index_fa:
    input:
        "{ref}.fa",
    output:
        "{ref}.fa.fai",
    shell:
        '''
        samtools faidx {input}
        '''
```

# The expand function - 2

```
rule get_region:
    input:
        fa="{ref}.fa",
        fai="{ref}.fa.fai",
    output:
        "{ref}.sub-{region}.fa"
    shell:
        '''
        samtools faidx {input.fa} {wildcards.region} > {output}
        '''
```

# The expand function - 2

```
rule run:
    input:
        expand(
            "{ref}.sub-{region}.fa.fai",
            ref=["hs37d5", "GRCh38"],
            region=["1:100-200", "1:300:400", "2:500-700"],
        )
```

# The expand function - 2

```
rule run:
    input:
        expand(
            "{ref}.sub-{region}.fa.fai",
            ref=["hs37d5", "GRCh38"],
            region=["1:100-200", "1:300:400", "2:500-700"],
        )
```

# The expand function - 2

```
rule run:
    input:
        expand(
            "{ref}.sub-{region}.fa.fai",
            ref=["hs37d5", "GRCh38"],
            region=["1:100-200", "1:300:400", "2:500-700"],
        )
```

