

exam1

```
#!/bin/bash

if [ $# -ne 2 ];
then
    exit 1;
fi

gender=$1;
age=$2;
if [ $age -lt 0 ];
then
    exit 1;
fi

if [ $gender = 'man' ];
then
    if [ $age -lt 20 ];
    then
        echo 'Man:Child';
    elif [ $age -lt 60 ];
    then
        echo 'Man:Adult';
    else
        echo 'Man:Elderly';
    fi
elif [ $gender = 'woman' ];
then
    if [ $age -lt 20 ];
    then
        echo 'Woman:Child';
    elif [ $age -lt 60 ];
    then
        echo 'Woman:Adult';
    else
        echo 'Woman:Elderly';
    fi
fi
```

exam2

```
#!/bin/bash

ls

read -p 'ファイル名を入力してください: ' file_name
read -p 'ファイルに追記する値を入力してください: ' input_val

if [ -f $file_name ];
then
    echo $input_val >> $file_name;
else
    echo 'ファイルが存在しません';
fi
```

exam3

```
#!/bin/bash

for i in `seq 1 100`;
do
    if [ $(( i % 3 )) -eq 0 ] && [ $(( i % 5 )) -eq 0 ];
    then
        echo $i FizzBuzz
    elif [ $(( i % 3 )) -eq 0 ];
    then
        echo $i Fizz
    elif [ $(( i % 5 )) -eq 0 ];
    then
        echo $i Buzz
    else
        echo $i
    fi
done
```

```
#!/bin/bash

i=1
while (( $i <= 100 ));
do
    if [ $(( i % 3 )) -eq 0 ] && [ $(( i % 5 )) -eq 0 ];
    then
        echo $i FizzBuzz
    elif [ $(( i % 3 )) -eq 0 ];
    then
        echo $i Fizz
    elif [ $(( i % 5 )) -eq 0 ];
    then
        echo $i Buzz
    else
        echo $i
    fi
    i=$(( i + 1 ))
done
```

```
#!/bin/bash

i=1
until (( $i > 100 ));
do
    if [ $(( i % 3 )) -eq 0 ] && [ $(( i % 5 )) -eq 0 ];
    then
        echo $i FizzBuzz
    elif [ $(( i % 3 )) -eq 0 ];
    then
        echo $i Fizz
    elif [ $(( i % 5 )) -eq 0 ];
    then
        echo $i Buzz
    else
        echo $i
    fi
    i=$(( i + 1 ))
done
```

exam4

```
#!/bin/bash

read -p 'ファイル名を入力してください: ' fh

if [ -f $fh ];
then
    read -p 'sum, avg, min, max, exit' command
    if [ $command = 'sum' ];
    then
        sum=0
        while read p;
        do
            sum=$(( sum + p ))
        done < $fh
        echo SUM: $sum
        exit 0
    elif [ $command = 'avg' ];
    then
        sum=0
        count=0
        while read p;
        do
            sum=$(( sum + p ))
            count=$(( count + 1 ))
        done < $fh
        echo AVG: $(( sum / count ))
        exit 0
    elif [ $command = 'min' ];
    then
        min=101
        while read p;
        do
            if [ $min -gt $p ];
            then
                min=$p
            fi
        done < $fh
        echo MIN: $min
        exit 0
    elif [ $command = 'max' ];
    then
        max=0
        while read p;
        do
            if [ $max -lt $p ];
            then
                max=$p
            fi
        done < $fh
        echo MAX: $max
```

```
        exit 0
    elif [ $command = 'exit' ];
    then
        exit 0
    else
        echo 'そのようなコマンドは存在しません'
        exit 1
    fi
else
    echo 'ファイルが存在しません'
    exit 1
fi
```

exam5

```
#!/bin/bash

function calculate_sum(){
    sum=0
    while read p;
    do
        sum=$(( sum + p ))
    done < $1
    echo SUM: $sum
    exit 0
}

function calculate_avg(){
    sum=0
    count=0
    while read p;
    do
        sum=$(( sum + p ))
        count=$(( count + 1 ))
    done < $1
    echo AVG: $(( sum / count ))
    exit 0
}

function calculate_min(){
    min=101
    while read p;
    do
        if [ $min -gt $p ];
        then
            min=$p
        fi
    done < $1
    echo MIN: $min
    exit 0
}

function calculate_max(){
    max=0
    while read p;
    do
        if [ $max -lt $p ];
        then
            max=$p
        fi
    done < $1
    echo MAX: $max
    exit 0
}
```



```
read -p 'ファイル名を入力してください: ' fh

if [ -f $fh ];
then
    read -p 'sum, avg, min, max, exit' command
    if [ $command = 'sum' ];
    then
        calculate_sum $fh
    elif [ $command = 'avg' ];
    then
        calculate_avg $fh
    elif [ $command = 'min' ];
    then
        calculate_min $fh
    elif [ $command = 'max' ];
    then
        calculate_max $fh
    elif [ $command = 'exit' ];
    then
        exit 0
    else
        echo 'そのようなコマンドは存在しません'
```

exam6

```
#!/bin/bash

select command in "list" "delete" "rename" "show" "exit"
do
    case $command in
        "list" )
            ls;;
        "delete")
            ls
            read -p "Enter file name you want to delete: " file_name
            if [ -f $file_name ];
            then
                rm $file_name
            fi;;
        "rename")
            ls
            read -p "Enter file name you want to rename: " file_name
            read -p "Enter new file name: " new_file_name
            if [ -f $file_name ];
            then
                mv $file_name $new_file_name
            fi;;
        "show")
            ls
            read -p "Enter file name you want to look: " file_name
            cat $file_name;;
        "exit")
            break;;
    esac
done
```

exam7

```
#!/bin/bash

if [ $# -ne 1 ];
then
    echo 'argument is wrong'
    exit 1
fi

function stop_exam7(){
    rm exam7.lock
    exit 0
}

if [ $1 = 'start' ];
then
    if [ -f 'exam7.lock' ];
    then
        echo 'Process is already running'
        exit 0
    else
        echo $$ > exam7.lock
        trap "stop_exam7" 2 15
        for i in `seq 1 1000`;
        do
            echo $i >> output_$.txt
            sleep 1
        done
        rm exam7.lock
        exit 0
    fi
elif [ $1 = 'stop' ];
then
    if [ -f 'exam7.lock' ];
    then
        PID=-1
        while read p;
        do
            PID=$p
        done < 'exam7.lock'
        kill -15 $PID
    else
        echo 'process is not running'
        exit 0
    fi
elif [ $1 = 'status' ];
then
    if [ -f 'exam7.lock' ];
    then
        PID=-1
        while read p;
```

```
do
    PID=$p
done < 'exam7.lock'
echo Process is running pid=$PID
else
    echo 'Process is not running'
fi
else
    echo 'wrong argument'
    exit 1
fi
```