



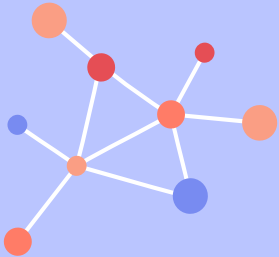
# BIOHACKING CON PYTHON

Marina Moro López



CIB3RTR4CKS

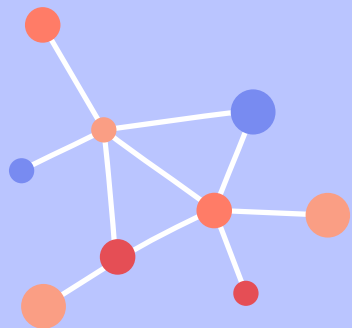
POWERED BY FUTURE SPACE

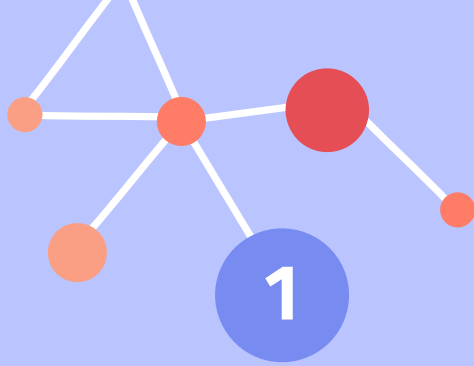


# ¡Hola! :D

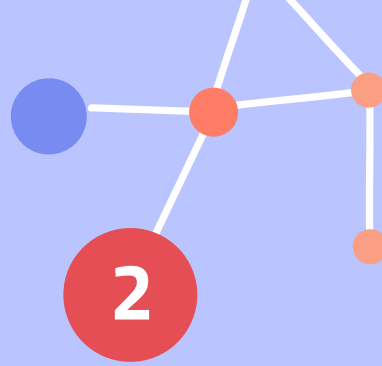
 @marinamorolopez  Marina Moro López

- Ingeniera biomédica
- Futura doctora en biofísica y bioingeniería
- Secretaria de Python España
- Programo en mi día a día científico

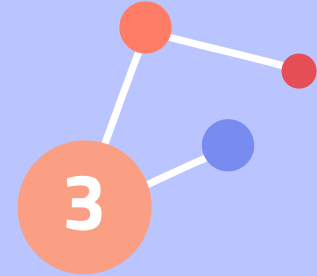




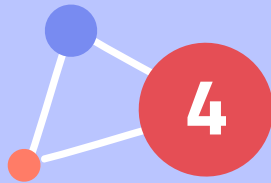
**DEFINICIÓN DE  
BIOHACKING**



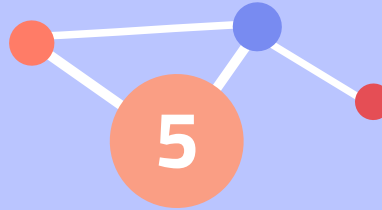
**TIPOS DE  
BIOHACKING**



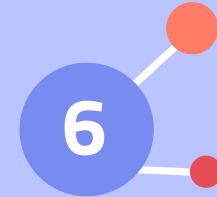
**TEORÍA BÁSICA  
DE GENÉTICA**



**CASO PRÁCTICO**



**CONSIDERACIONES  
IMPORTANTES**

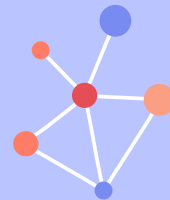
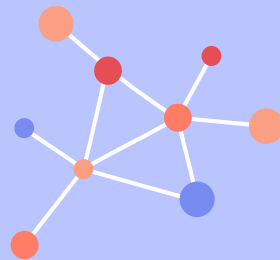
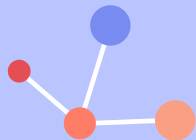


**RONDA DE  
PREGUNTAS**



1

# DEFINICIÓN DE BIOHACKING





# Bio + hacking

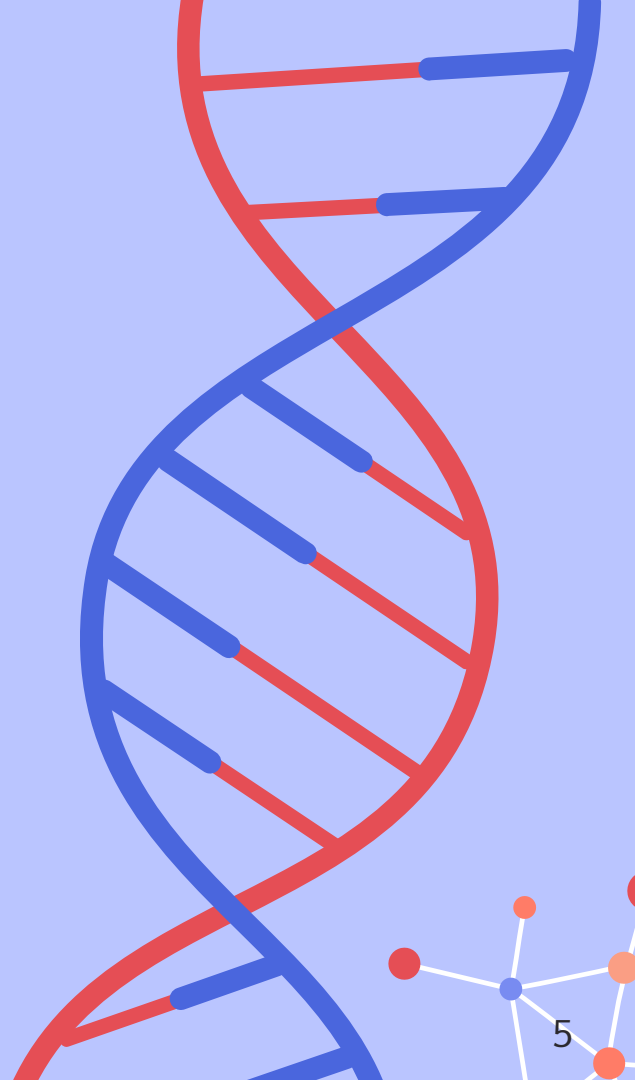
Añadir funcionalidades y resolución de problemas sociales en el ámbito bio

## Democratización

Herramientas biológicas e información fuera del ámbito institucional

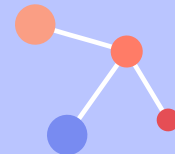
## Código ético

Transparencia, seguridad, educación, compromiso y responsabilidad





# ORIGEN DEL BIOHACKING



## DIY

Autosuficiencia y  
comunidad

## TRANSHUMANISMO

Superación de límites  
biológicos con tecnología



## HACKING

Democratización y hackeo  
de procedimientos

# ¿Y QUÉ SE HACE?

Proyectos de salud, medioambiente y bioarte usando tecnologías de genética, bioquímica, bioingeniería, biología sintética, electrónica...

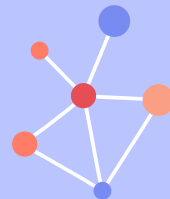
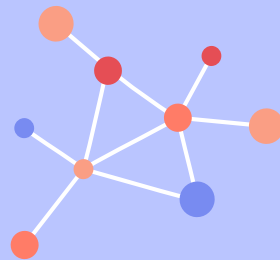
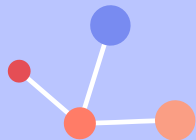
- Desarrollo de equipos low-cost
- Producción de medicamentos
- Talleres y conferencias
- Start-ups con los productos desarrollados
- Autoexperimentación y modificaciones corporales



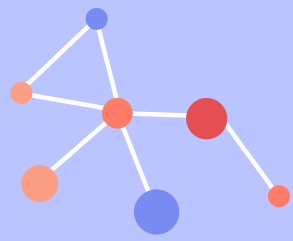


2

## TIPOS DE BIOHACKING





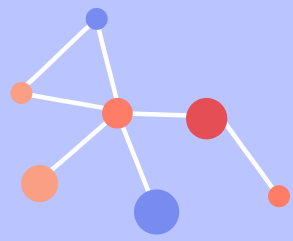


# BIOHACKING FISIOLÓGICO

Hacking del propio organismo  
con dietas, ingesta de  
suplementos y hábitos de vida

**Ejemplos:** ayuno intermitente,  
exposición a infrarrojos, uso  
de nootrópicos



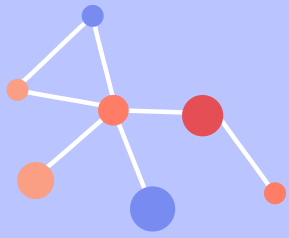


# BIOLOGÍA DIY

Manipulación de la biología a través de técnicas innovadoras por parte de la ciudadanía

**Ejemplos:** biohacking genético, neurohacking, terapia celular, producción de medicamentos y de equipo



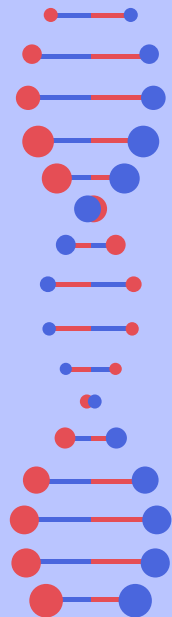


# GRINDER

Manipulación corporal con  
visión transhumanista (body  
hacking)

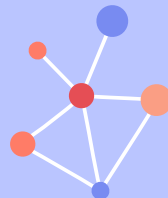
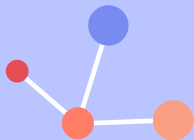
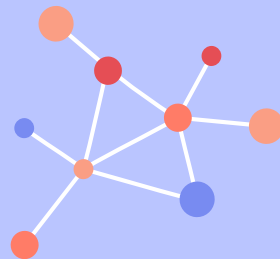
**Ejemplos:** implantación de  
hardware, modificación de  
implantes, edición genética y  
biohacking *in vivo*





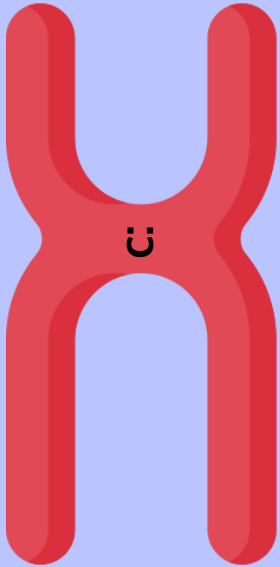
3

# TEORÍA BÁSICA DE GENÉTICA



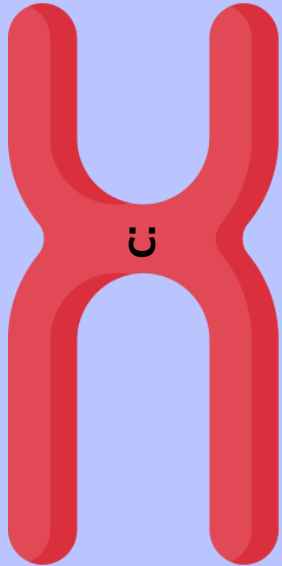
# CROMOSOMA – GEN – ADN

Estructura que contiene  
todos los genes

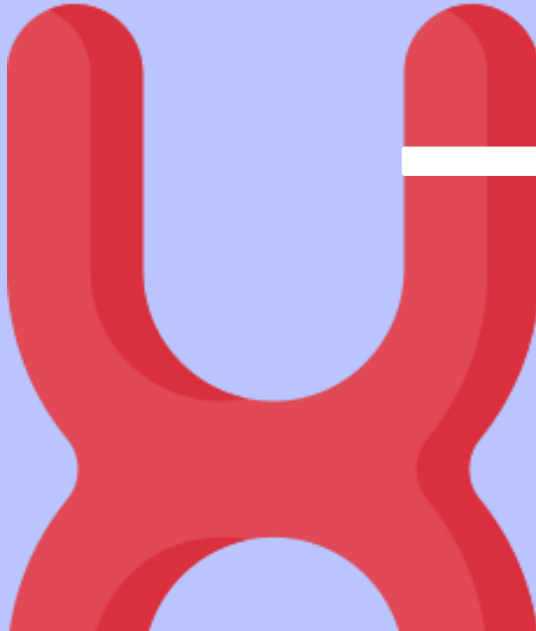


# CROMOSOMA – GEN – ADN

Estructura que contiene  
todos los genes

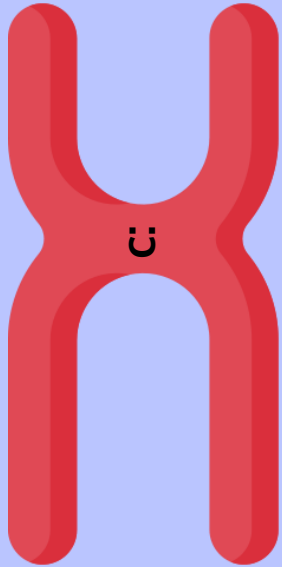


Segmento de ADN que  
determina un rasgo

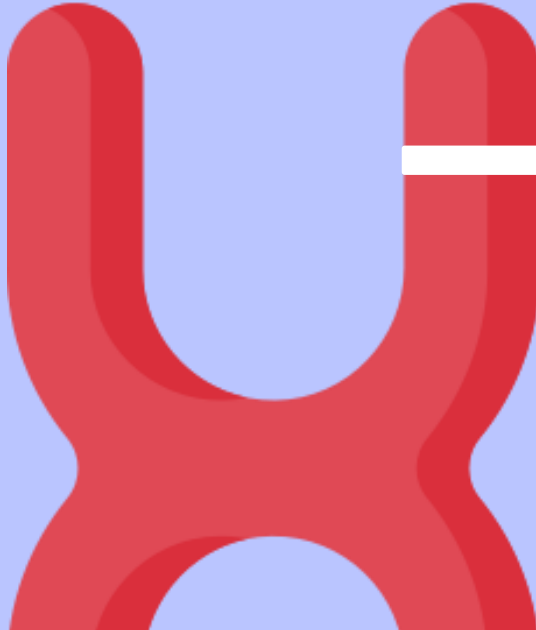


# CROMOSOMA – GEN – ADN

Estructura que contiene  
todos los genes



Segmento de ADN que  
determina un rasgo



Doble hélice  
formada por bases



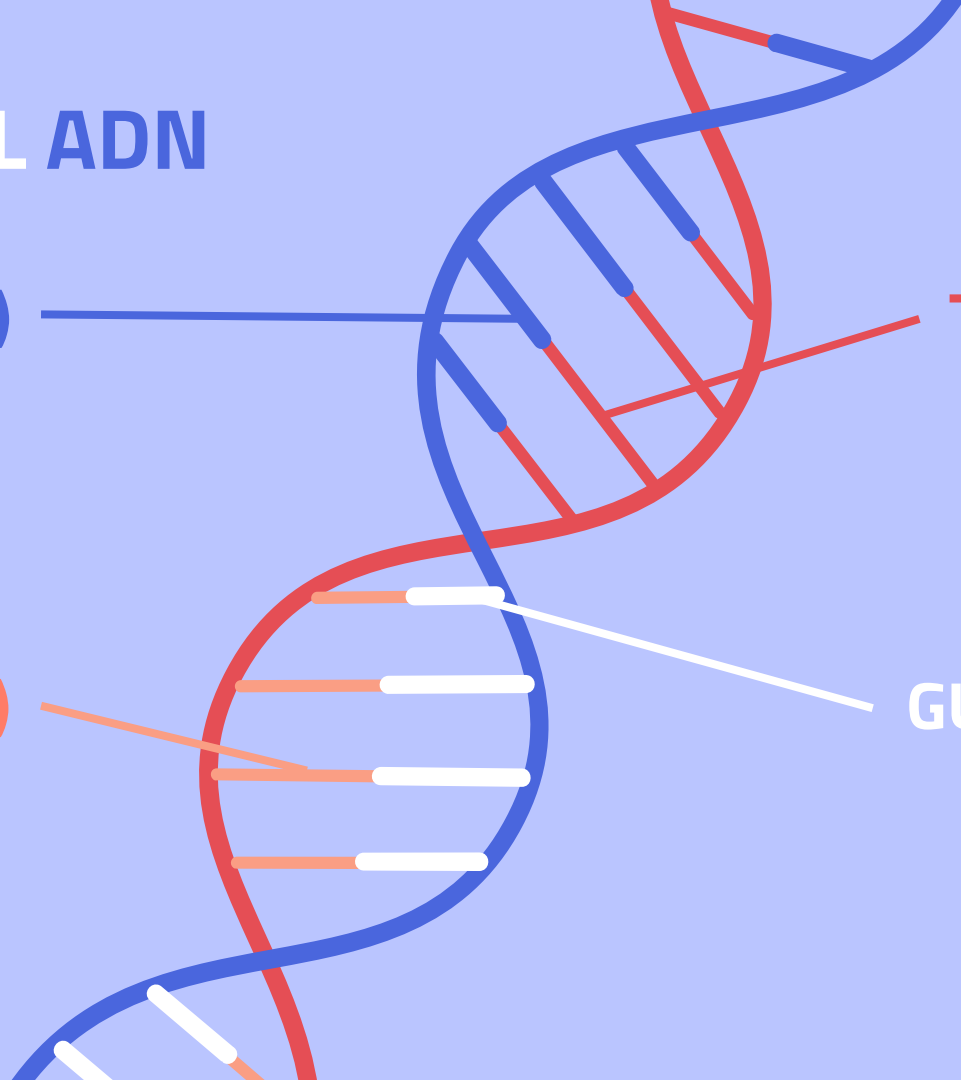
# **BASES DEL ADN**

**ADENINA (A)**

**TIMINA (T)**

**CITOSINA (C)**

**GUANINA (G)**

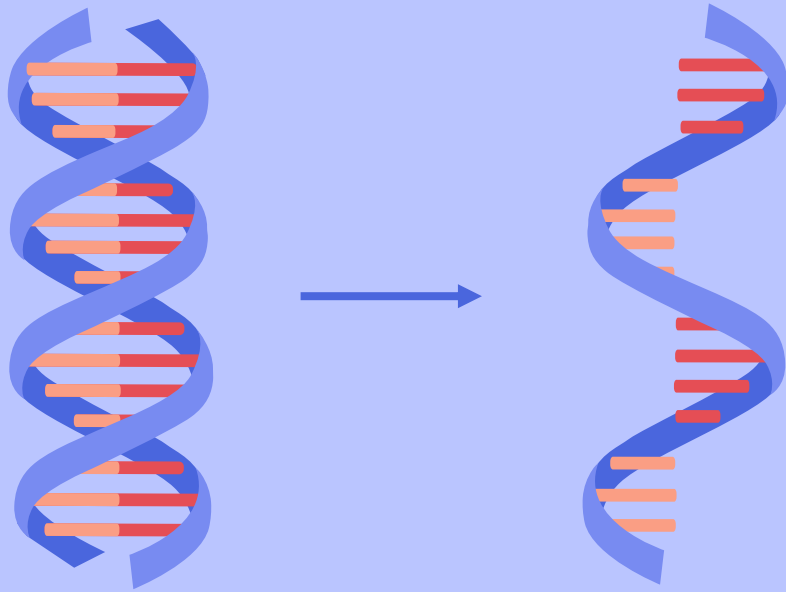




# ADN – ARN – PROTEÍNA

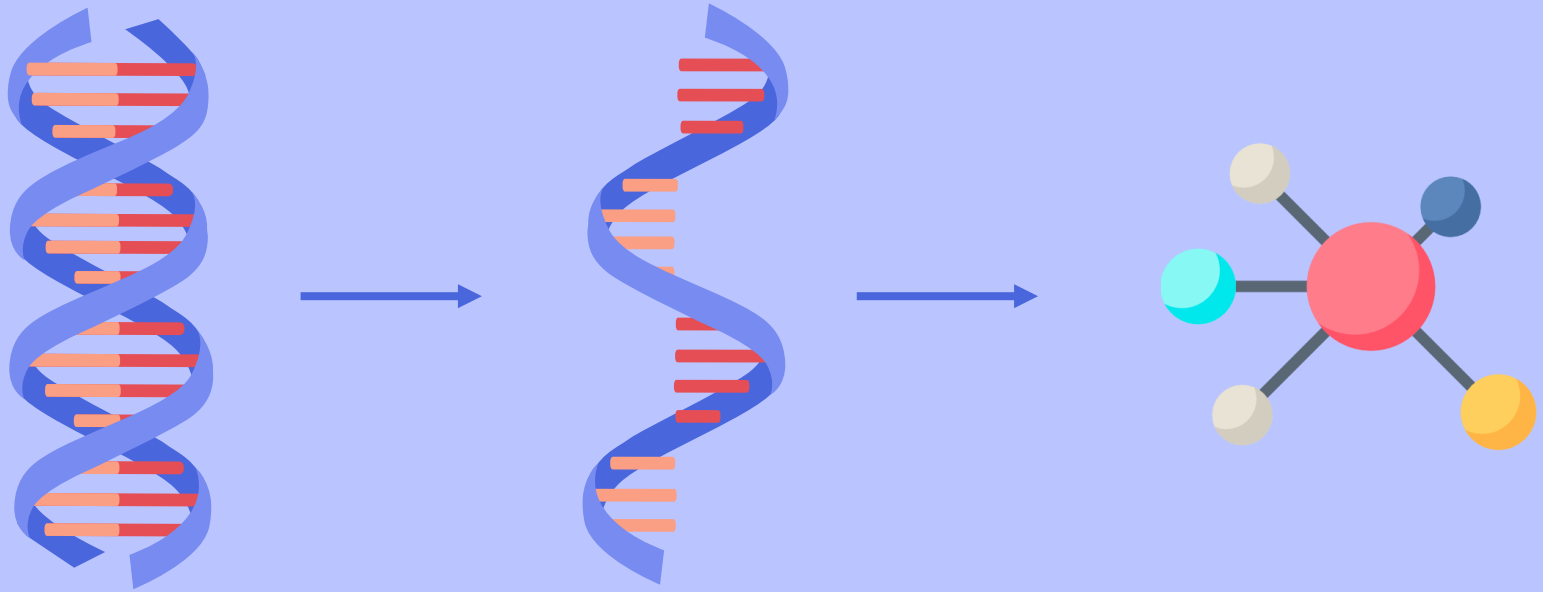


# ADN – ARN – PROTEÍNA



Hélice simple  
 $T > U$

# ADN – ARN – PROTEÍNA



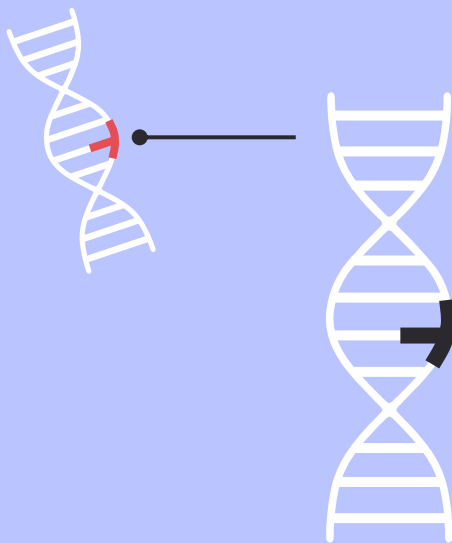
Hélice simple  
 $T > U$

# MUTACIONES DEL ADN



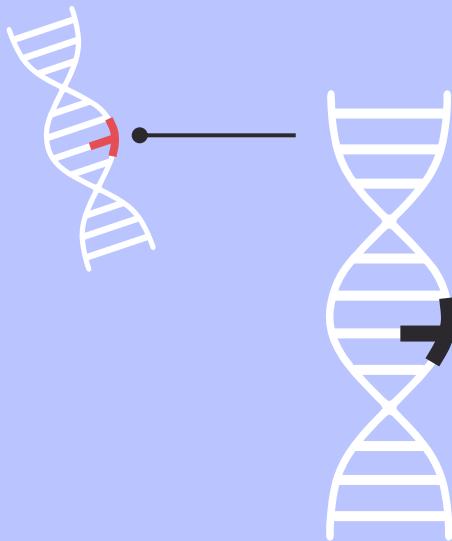
# MUTACIONES DEL ADN

**SUSTITUCIÓN  
KNOCK-IN**

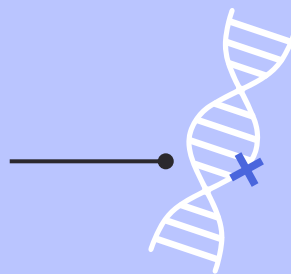


# MUTACIONES DEL ADN

**SUSTITUCIÓN  
KNOCK-IN**



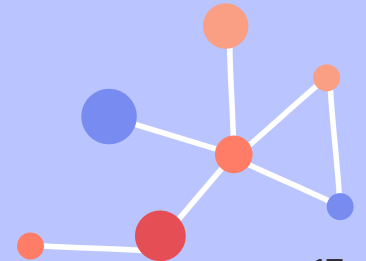
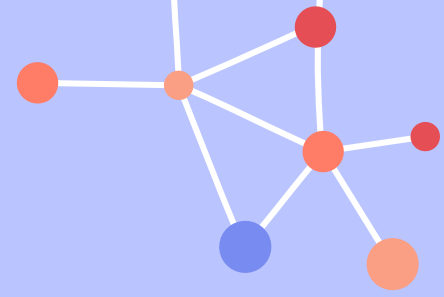
**DELECIÓN  
KNOCK-OUT**



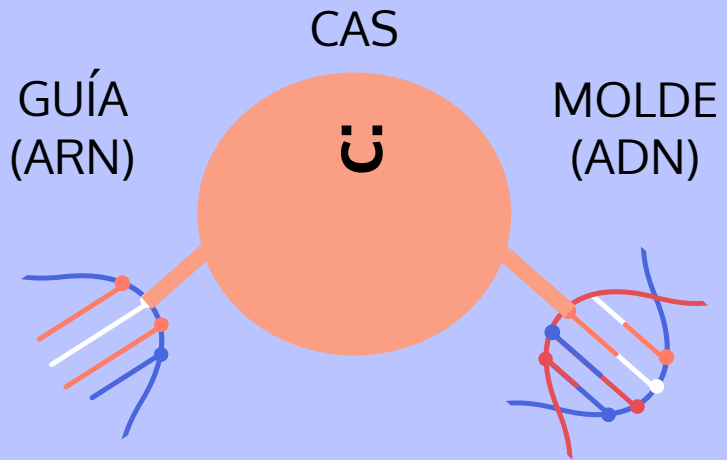


# CRISPR

Corta y pega de secuencias  
de ADN (edición genética)

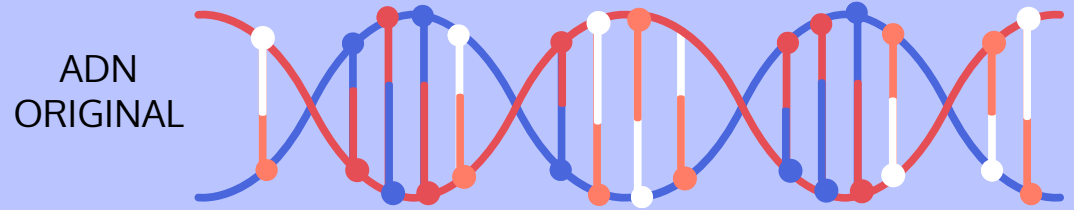
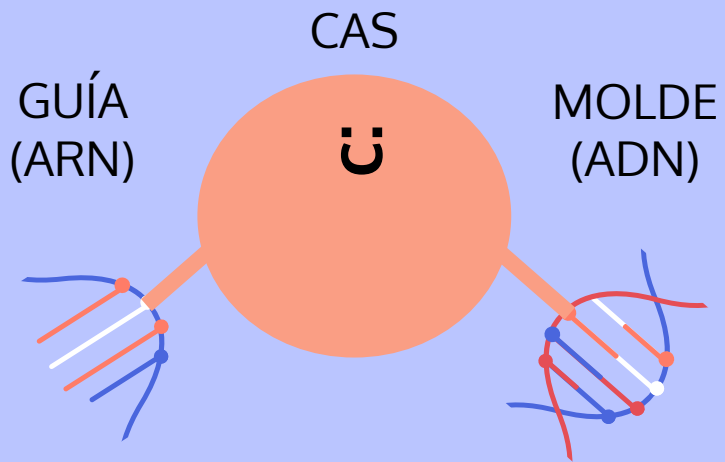


# CRISPR

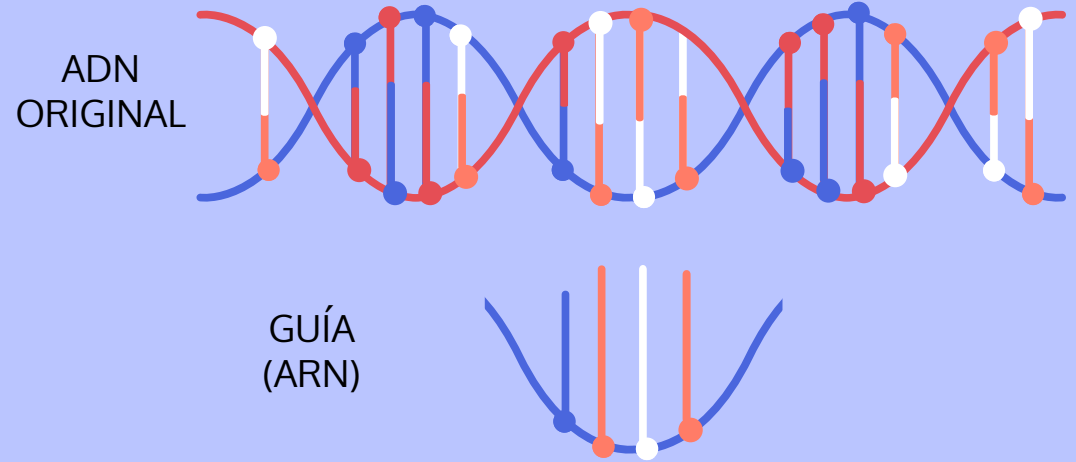
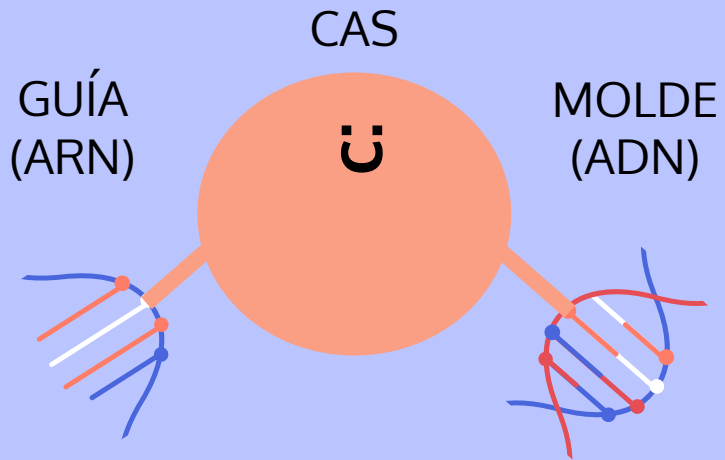




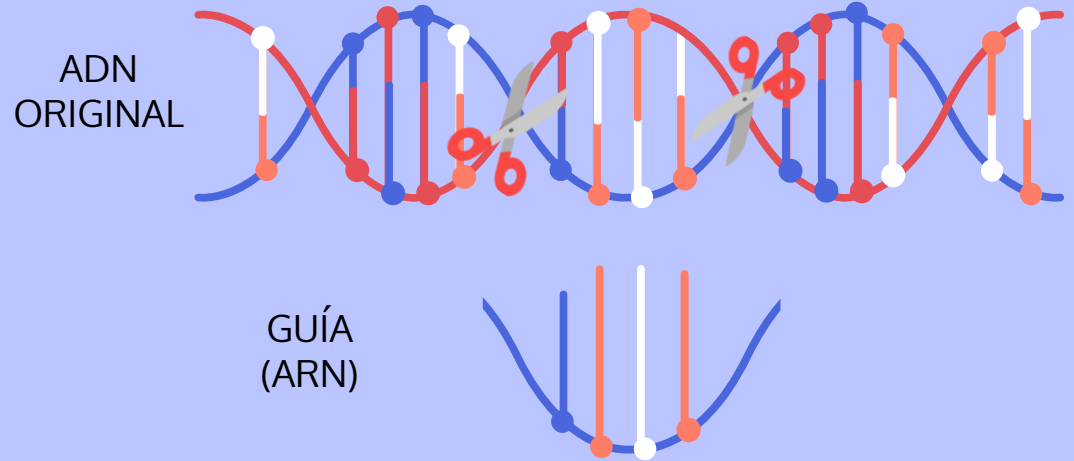
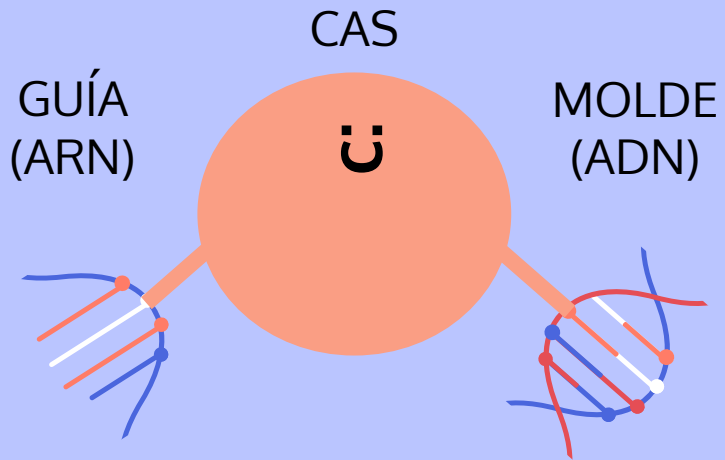
# CRISPR



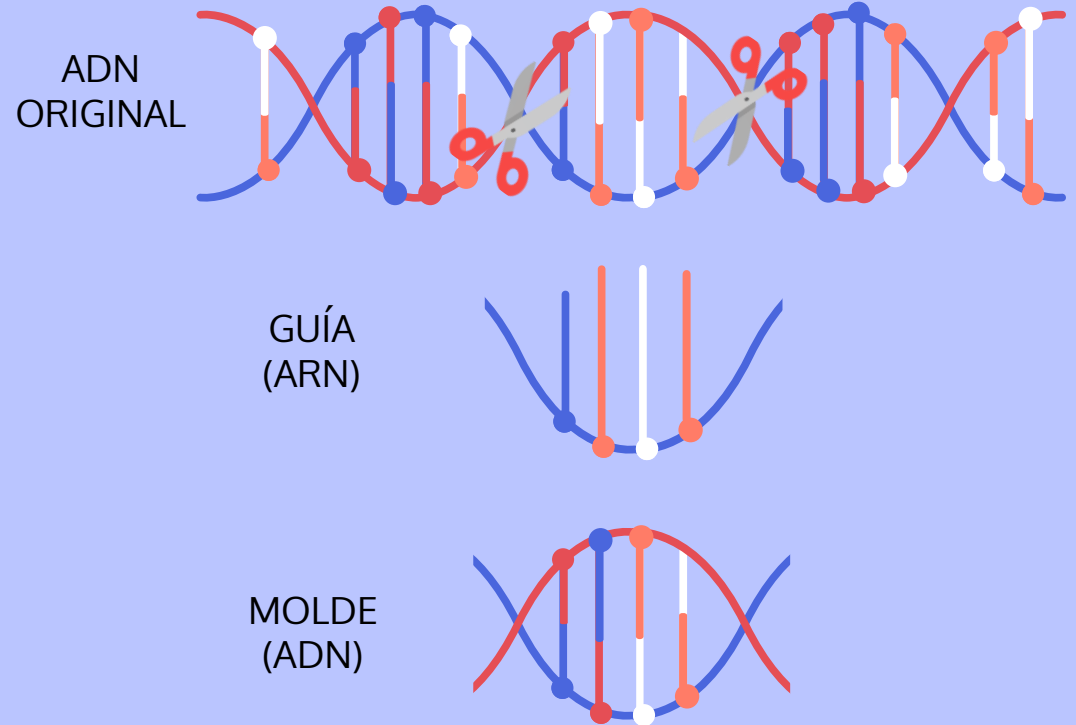
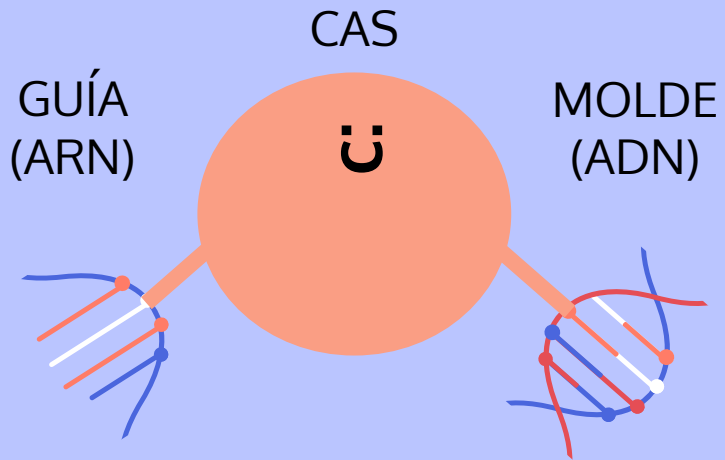
# CRISPR



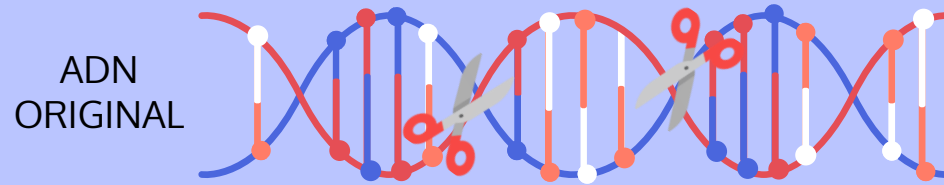
# CRISPR

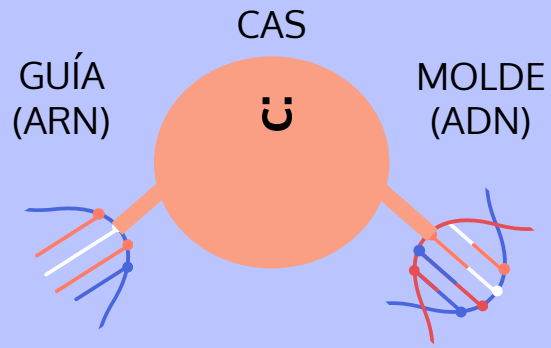


# CRISPR

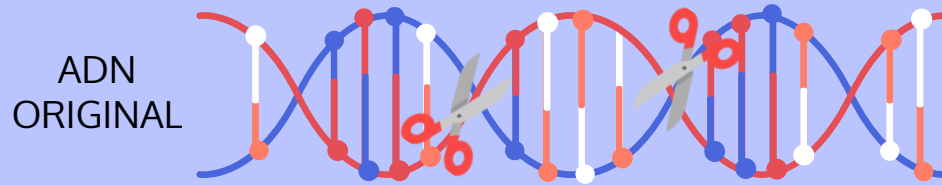


# CRISPR

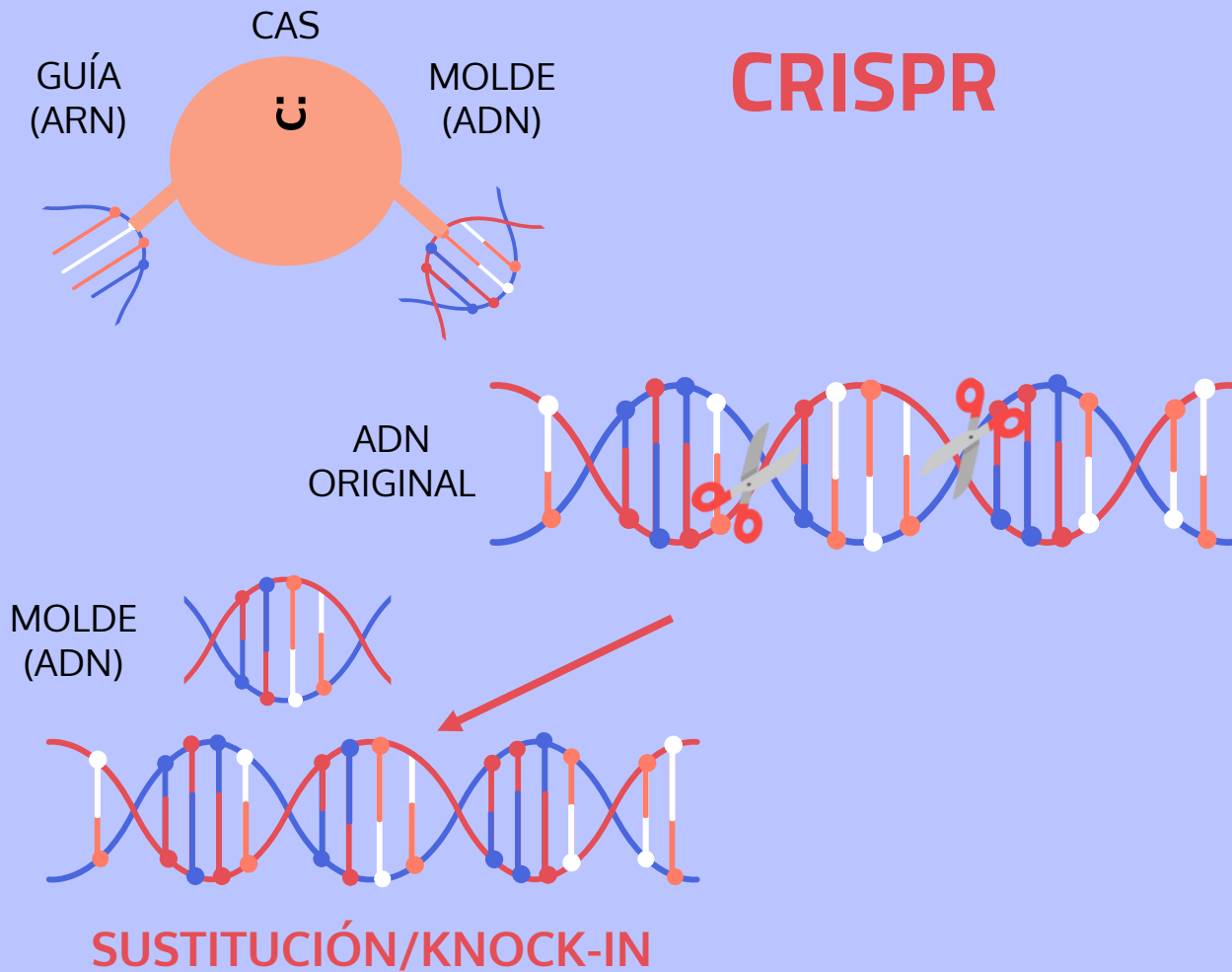




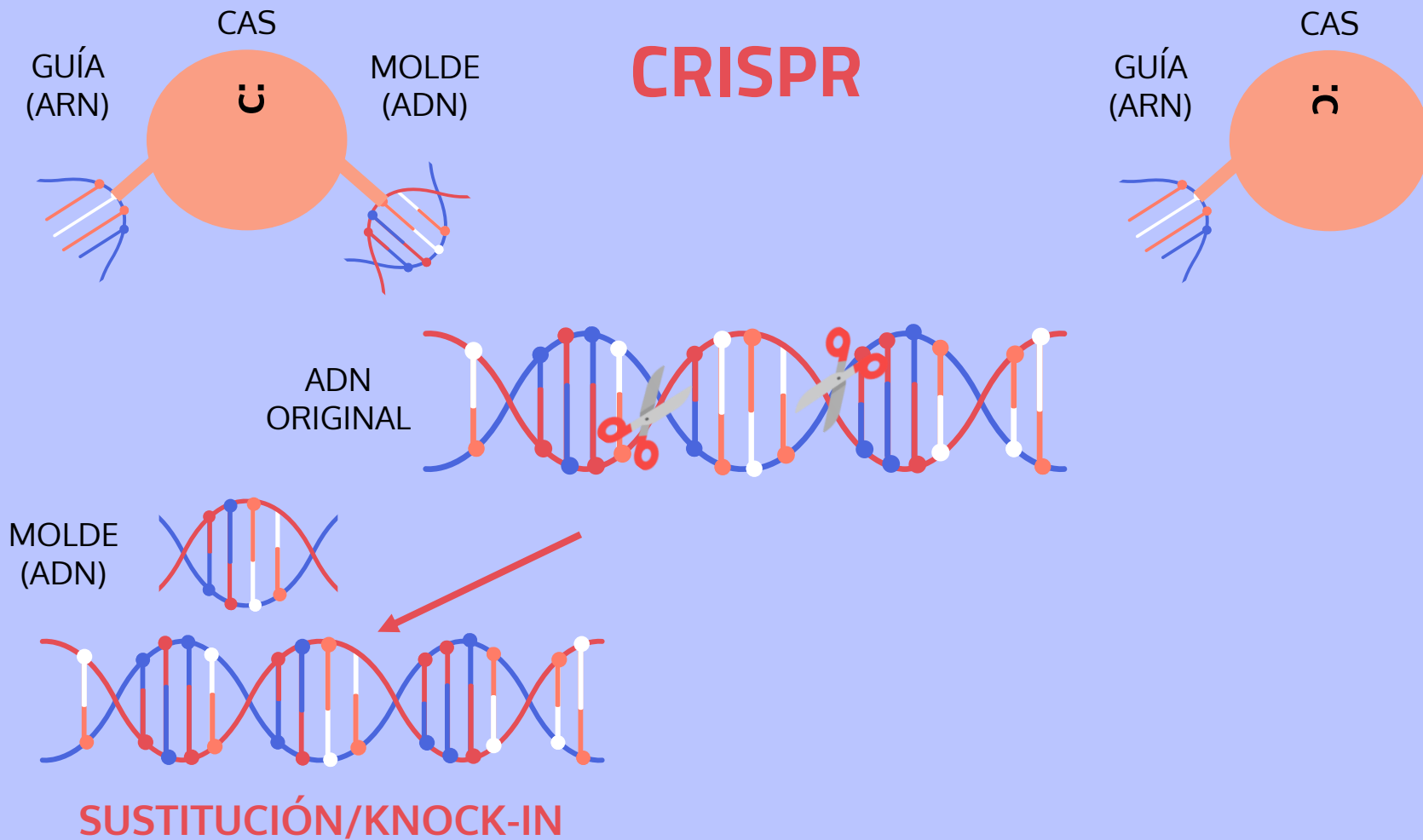
# CRISPR



# CRISPR

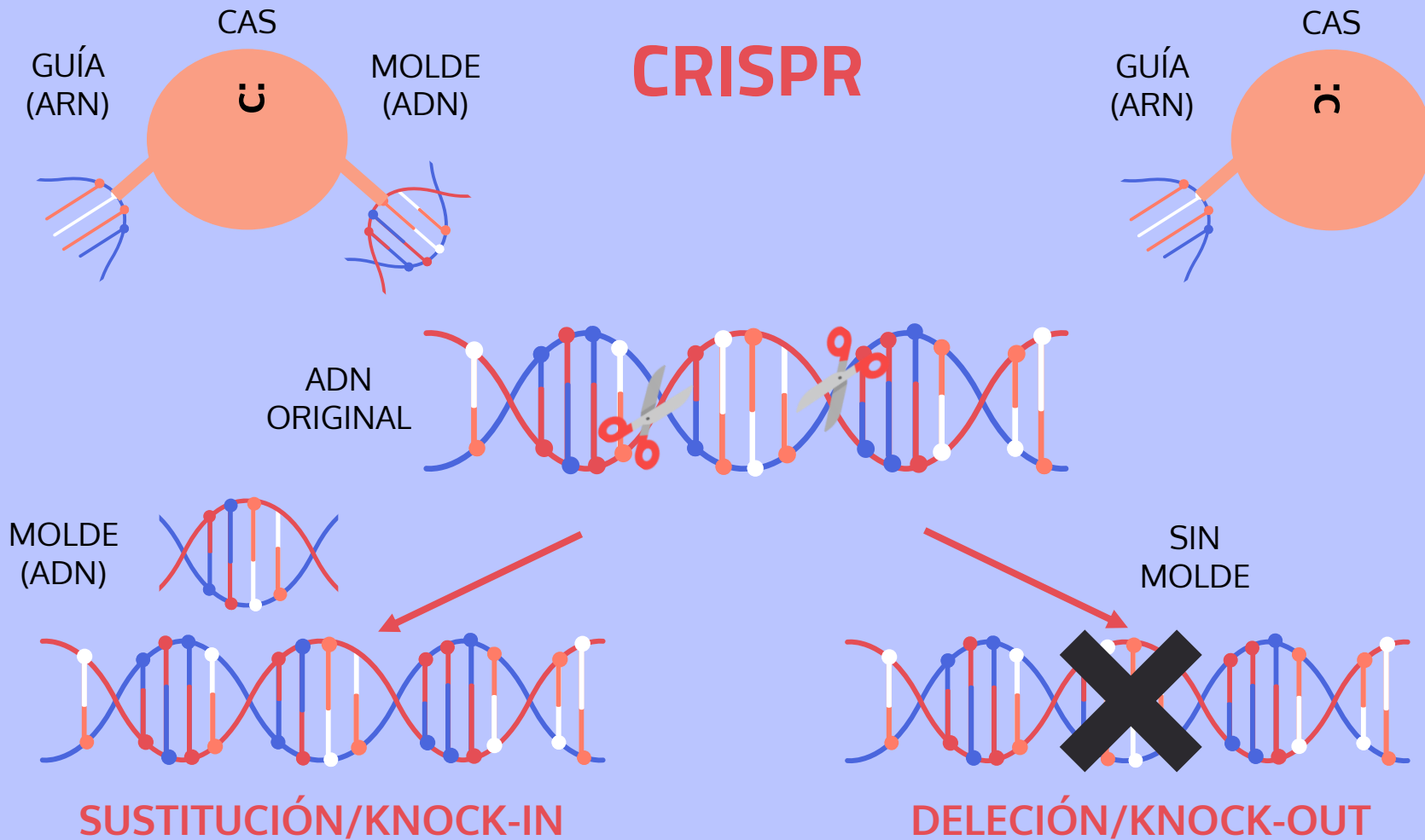


# CRISPR





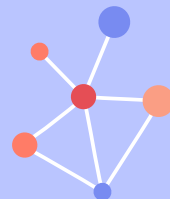
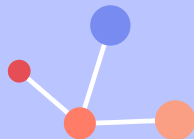
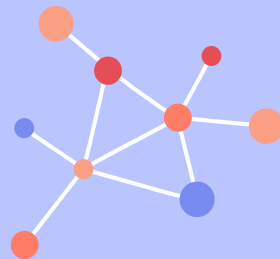
# CRISPR





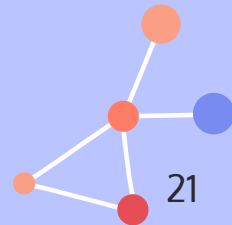
4

# CASO PRÁCTICO





# Cómo convertirnos en el Sr Burns fluorescente



# Cómo convertirnos en el Sr Burns fluorescente



1

Calvicie y pelo azul-canoso

2

Bioluminiscencia

3

Pupilas dilatadas

# Cómo convertirnos en el Sr Burns fluorescente



1

Calvicie y pelo azul-canoso →

Delección del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)

2

Bioluminiscencia

3

Pupilas dilatadas

# Cómo convertirnos en el Sr Burns fluorescente



1

Calvicie y pelo azul-canoso →

Delección del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)

2

Bioluminiscencia →

Sustitución de GFP al final del gen TYR

3

Pupilas dilatadas

# Cómo convertirnos en el Sr Burns fluorescente



- 1 Calvicie y pelo azul-canoso → Deleción del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)
- 2 Bioluminiscencia → Sustitución de GFP al final del gen TYR
- 3 Pupilas dilatadas → Sustancias pecaminosas / ciclopentolato

```

from tkinter.filedialog import askopenfile

def main():

    gene_file = askopenfile(mode='r')
    gene_seq = gene_file.readlines()[1:]
    gene_seq = ''.join(gene_seq).replace('\n', '')

    mutation_type = input("Introduce mutation type (in/out): ")

    while mutation_type != 'in' and mutation_type != 'out':
        print('Invalid input.')
        mutation_type = input("Introduce mutation type (in/out): ")

    if mutation_type == 'in':
        knockin_type = input("Introduce the knock-in position in the gene (mid/end): ")
        while knockin_type != 'mid' and knockin_type != 'end':
            print('Invalid input.')
            knockin_type = input("Introduce the knock-in position in the gene (mid/end): ")

        if knockin_type == "mid":
            DNA_guide, mutated_gene_seq, mold = knock_in_mid(gene_seq)
        elif knockin_type == "end":
            DNA_guide, mutated_gene_seq, mold = knock_in_end(gene_seq)

    else:
        DNA_guide, mutated_gene_seq, mold = knock_out(gene_seq)

```



marinamorolopez / biohacking-c1b3rtr4cks24



```

from tkinter.filedialog import askopenfile

def main():

    gene_file = askopenfile(mode='r')
    gene_seq = gene_file.readlines()[1:]
    gene_seq = ''.join(gene_seq).replace('\n', '')

    mutation_type = input("Introduce mutation type (in/out): ")

    while mutation_type != 'in' and mutation_type != 'out':
        print('Invalid input.')
        mutation_type = input("Introduce mutation type (in/out): ")

    if mutation_type == "in":

        knockin_type = input("Introduce the knock-in position in the gene (mid/end): ")
        while knockin_type != 'mid' and knockin_type != 'end':
            print('Invalid input.')
            knockin_type = input("Introduce the knock-in position in the gene (mid/end): ")

        if knockin_type == "mid":
            DNA_guide, mutated_gene_seq, mold = knock_in_mid(gene_seq)
        elif knockin_type == "end":
            DNA_guide, mutated_gene_seq, mold = knock_in_end(gene_seq)

    else:
        DNA_guide, mutated_gene_seq, mold = knock_out(gene_seq)

```

```
mutated_gene_file = open('MUTATED_SEQUENCE.txt', 'w')
mutated_gene_file.write(mutated_gene_seq)
mutated_gene_file.close()

guide_file = open('GUIDE.txt', 'w')
guide_file.write(DNA_to_RNA(DNA_guide))
guide_file.close()

mold_file = open('MOLD.txt', 'w')
mold_file.write(mold)
mold_file.close()
```

```
def DNA_to_RNA(DNA_guide):

    RNA_guide = ""
    for base in DNA_guide:
        if base == "T":
            RNA_guide += "A"
        elif base == "A":
            RNA_guide += "U"
        elif base == "C":
            RNA_guide += "G"
        elif base == "G":
            RNA_guide += "C"

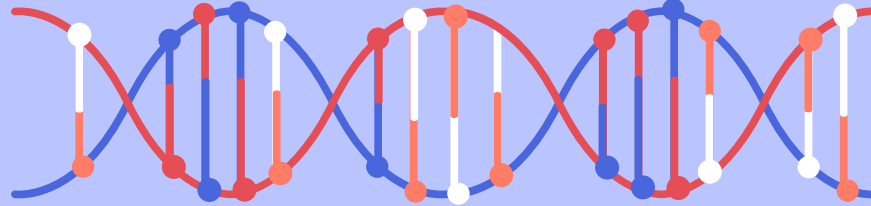
    return RNA_guide
```

1

Calvicie y pelo azul-canoso →

Delección del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)

GEN IRF4  
ORIGINAL



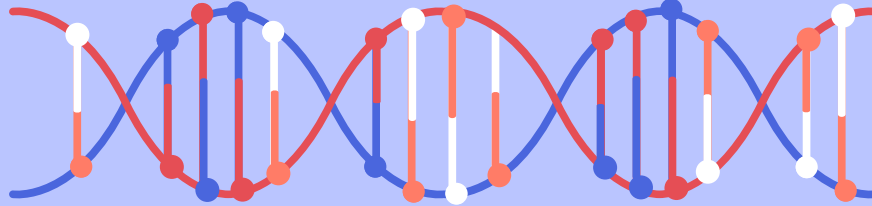
GEN IRF4  
MUTADO

1

Calvicie y pelo azul-canoso →

Delección del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)

GEN IRF4  
ORIGINAL



GEN IRF4  
MUTADO



1

Calvicie y pelo azul-canoso →

Delección del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)

```
def knock_out(gene_seq):  
    DNA_guide = gene_seq  
    mutated_gene_seq = ""  
    mold = ""  
  
    return DNA_guide, mutated_gene_seq, mold
```

1

Calvicie y pelo azul-canoso →

Delección del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)

ADN ORIGINAL

```
GTGTCATTCCCCATCCTGGAAACCCTCCAGCAACCCCTGACTCCCCGACCGCCCCACCCCCTGCCGAGCA  
CGTCTACTCAGCCCCATAACTGCTTGTCTTCCCTCCTCTGCCACCCATGCACCTGCCCCGTCTGAGAGCT  
CTCTACCTCACCCCGAGGCCTCCCCGGCCTCCTGGCCATTGTTCTCTCTCGGGCGGTACCCACACTATGG  
CCAGATAATTCTCCTTTACTGTAGTTCTTACCTTATTACGGGGGAATATGAGCCAAAGCCATGTAAC TC
```

GUÍA (ARN)

```
CACAGUAAGGGGUAGGACCUUUGGGAGGUCGUUGGGGACUGAGGGGCUGGCGGGGUGGGGGACGGCUCGU  
GCAGAUGAGUCGGGGUAUUGACGAACAGGAAGGGAGGAGACGGUGGGUACGUGGACGGGCAGACUCUCGA  
GAGAUGGAGUGGGGCUCCGGAGGGGCCGGAGGACCGGUAACAAGAGAGAGCCCGCCAUGGGUGUGAUACC  
GGUCUAUUAAGGAGGAAAUGACAUCAAGAAUGGAAUAAUGCCCCCUUAUACUCGGUUUCGGUACA AUAG
```

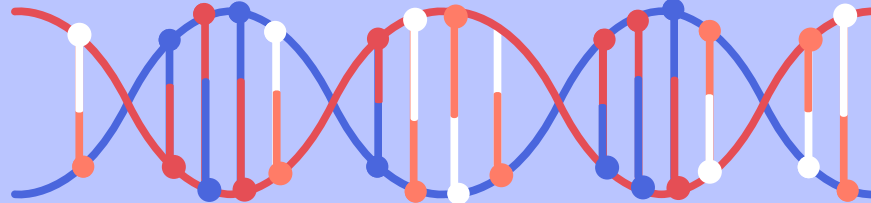
SIN MOLDE DE ADN

1

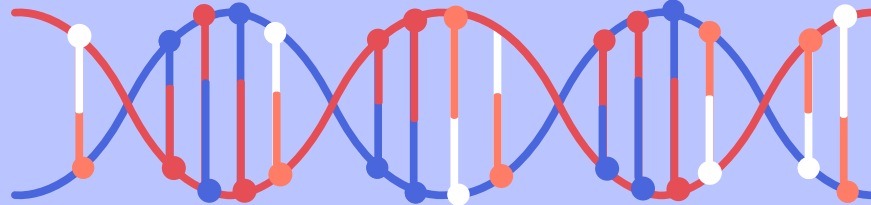
Calvicie y pelo azul-canoso →

Delección del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)

GEN AR  
ORIGINAL



GEN AR  
MUTADO



1

Calvicie y pelo azul-canoso →

Delección del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)

```
def knock_in_mid(gene_seq):  
  
    mutation_position = int(input("Introduce the numeric position of the mutation base (e.g. 1, 25, 203): "))  
    while mutation_position <= 0:  
        print('Invalid input. Introduce positive integer. ')  
        mutation_position = int(input("Introduce the numeric position of the mutation base (e.g. 1, 25, 203): "))  
  
    mutation_base = input("Introduce the new base corresponding to the defined mutation position in upper case (A/T/G/C): ")  
    while mutation_base != 'A' and mutation_base != 'T' and mutation_base != 'G' and mutation_base != 'C':  
        print('Invalid input. ')  
        mutation_base = input("Introduce the new base corresponding to the defined mutation position (A/T/G/C): ")  
  
    DNA_guide = gene_seq[mutation_position-25:mutation_position+25]  
    mutated_gene_seq = gene_seq[:mutation_position-1] + mutation_base + gene_seq[mutation_position:]  
    mold = mutated_gene_seq[mutation_position-25:mutation_position+25]  
  
    return DNA_guide, mutated_gene_seq, mold
```



1

Calvicie y pelo azul-canoso →

Delección del gen IRF4 completo  
Sustitución mitad del gen AR (363, C>T)

ADN ORIGINAL

ATGCTAGCTGCCGTTTTGTGTTATCTGTTACAGACTAATACAATTTGCAA

GUÍA (ARN)

UACGAUCGACGGCAAAACACAAUAGACAAUGUCUGAUUAUGUUAACGUU

MOLDE (ADN)

ATGCTAGCTGCCGTTTTGTGTTATTGTTACAGACTAATACAATTTGCAA

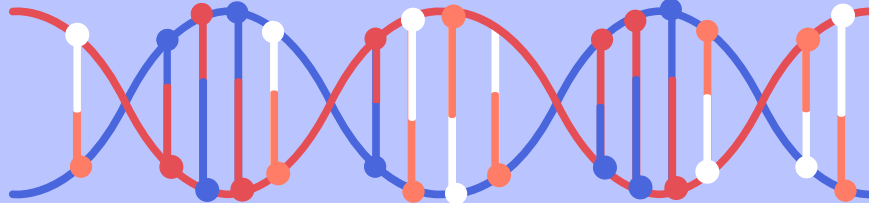
2

Bioluminiscencia

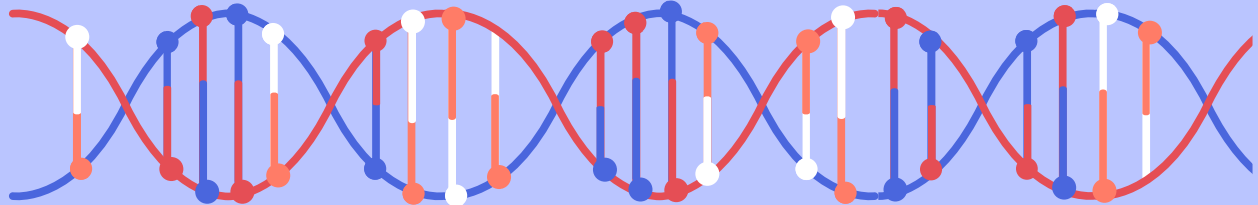


Sustitución de GFP al final del gen TYR

GEN TYR  
ORIGINAL



GEN TYR  
MUTADO



2

Bioluminiscencia



Sustitución de GFP al final del gen TYR

```
def knock_in_end(gene_seq):  
  
    plasmid_file = askopenfile(mode='r')  
    plasmid_seq = plasmid_file.readlines()[1:]  
    plasmid_seq = ''.join(plasmid_seq).replace('\n', '')  
  
    DNA_guide = gene_seq[len(gene_seq)-50:len(gene_seq)]  
    mutated_gene_seq = gene_seq + plasmid_seq  
    mold = DNA_guide + plasmid_seq  
  
    return DNA_guide, mutated_gene_seq, mold
```

2

Bioluminiscencia



Sustitución de GFP al final del gen TYR

FINAL DE TYR

TTTGAAGGCAAGATTTTAGTCCAGGGTGGTATGTTTCAATCA|TTTTTGCA

GUÍA (ARN)

|AAACUCCGUUCUAAAAUCAGGUCCACCAUACAAAGUUAGUAAAAACGU

MOLDE (ADN)

TTTGAAGGCAAGATTTTAGTCCAGGGTGGTATGTTTCAATCATT|TTTTTGCAATGGTGAGCAAGGGCGAGGAGCTGTTC  
 ACCGGGGTGGTGCCATCCTGGTCGAGCTGGACGGCGACGTAAACGGCCACAAGTTCAGCGTGTCCGGCGAGGGCGA  
 GGGCGATGCCACCTACGGCAAGCTGACCCTGAAGTTCATCTGCACCACCGGCAAGCTGCCC GTGCCCTGGCCACCC  
 TCGTGACCACCCTGACCTACGGCGTGCAGTGCTTCAGCCGCTACCCCGACCACATGAAGCAGCACGACTTCTTCAAG  
 TCCGCCATGCCCCGAAGGCTACGTCCAGGAGCGCACCATCTTCTTCAAGGACGACGGCAACTACAAGACCCGCGCCGA  
 GGTGAAGTTCGAGGGGCGACACCCTGGTGAACCGCATCGAGCTGAAGGGCATCGACTTCAAGGAGGACGGCAACATCC  
 TGGGGCACAAGCTGGAGTACAAC TACAACAGCCACAACGTCTATATCATGGCCGACAAGCAGAAGAACGGCATCAAG  
 GTGAACTTCAAGATCCGCCACAACATCGAGGACGGCAGCGTGCAGCTCGCCGACCACTACCAGCAGAACACCCCAT  
 CGGCGACGGCCCCGTGCTGCTGCCCGACAACCACTACCTGAGCACCCAGTCCGCCCTGAGCAAAGACCCCAACGAGA  
 AGCGCGATCACATGGTCCTGCTGGAGTTCGTGACCGCCGCCGGGATCACTCTCGGCATGGACGAGCTGTACAAG

3

Pupilas dilatadas



Sustancias pecaminosas  
Ciclopentolato





## Comprar kit CRISPR DIY



1

Comprar kit CRISPR DIY

2

Preparar las tres jeringas (genes TYR, AR e IRF4) e inyectar



1

Comprar kit CRISPR DIY

2

Preparar las tres jeringas (genes TYR, AR e IRF4) e inyectar

3

Pupilas dilatadas con el método a vuestra elección





1

Comprar kit CRISPR DIY

2

Preparar las tres jeringas (genes TYR, AR e IRF4) e inyectar

3

Pupilas dilatadas con el método a vuestra elección

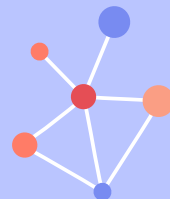
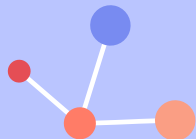
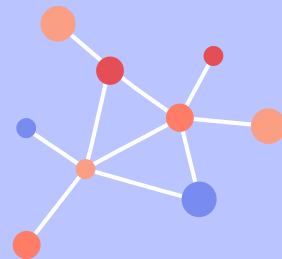


**¡Y listo!**



5

## CONSIDERACIONES IMPORTANTES



# RIESGOS Y LEGALIDAD



Condiciones y procedimientos peligrosos



Introducción de mutaciones en la línea germinal



Contaminación ambiental con agentes biológicos



Medicamentos no probados en el mercado

# RIESGOS Y LEGALIDAD



Condiciones y procedimientos peligrosos



Introducción de mutaciones en la línea germinal



Contaminación ambiental con agentes biológicos



Medicamentos no probados en el mercado



Código ético y auto-seguridad

# RIESGOS Y LEGALIDAD



Condiciones y procedimientos peligrosos



Introducción de mutaciones en la línea germinal



Contaminación ambiental con agentes biológicos



Medicamentos no probados en el mercado



Código ético y auto-seguridad



Legalidad dudosa

# RIESGOS Y LEGALIDAD



Condiciones y procedimientos policresos

In terminal

Co lógicos

M o

**DON'T TRY THIS AT HOME!**



Código ético y auto-seguridad



Legalidad dudosa





# APLICACIONES TERAPÉUTICAS



Edición de genes que producen enfermedades genéticas (epidermólisis bullosa, anemia de Falconi)



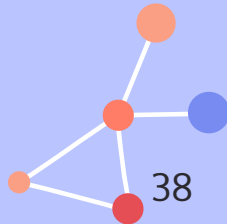
Células CAR-T contra el cáncer

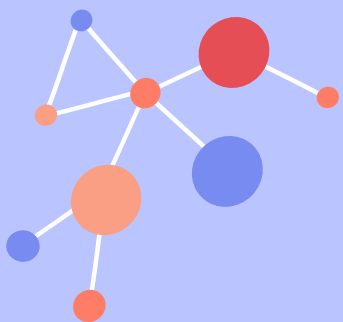


Terapia antiviral (SARS-CoV-2, VIH)



Lucha contra enfermedades infecciosas (malaria, fiebre amarilla)





# ¡Gracias!

## ¿Preguntas?



marinamorolopez@es.python.org



@marinamorolopez



Marina Moro López



marinamorolopez



marinamorolopez / biohacking-c1b3rtr4cks24



C1B3RTR4CKS

POWERED BY FUTURE SPACE



Python  
España