

API para generar código QR:

URL	Method
https://api.qrserver.com/v1/create-qr-code	GET

1. Argumento de entrada:

Campo	Tipo de dato	Observaciones
size	String	El tamaño debe indicarse en píxeles de una misma dimensión, por ejemplo 150x150
margin	int	Margen en píxeles
data	String	Información que tendrá el código QR

2. Argumento de salida:

- HTTP Status: 200 OK
- Ejemplo:

GET https://api.peruapis.co GET https://api.qrserver.com + ... No Environment

https://api.qrserver.com/v1/create-qr-code/?size=150x150&margin=7&data=4 Save


GET https://api.qrserver.com/v1/create-qr-code/?size=150x150&margin=7&data=4 Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/>	size	150x150			
<input checked="" type="checkbox"/>	margin	7			
<input checked="" type="checkbox"/>	data	4			
	Key	Value	Description		

Body Cookies Headers (9) Test Results Status: 200 OK Time: 254 ms Size: 735 B Save Response



- **HTTP Status: 400 Bad Request**
- **Ejemplo:**

The screenshot shows a REST client interface with the following details:

- URL:** `https://api.qrserver.com/v1/create-qr-code/?size=150x150&margin=7&data=`
- Method:** GET
- Status:** 400 Bad Request
- Time:** 330 ms
- Size:** 454 B
- Response Body (Text):**

```
1 en: malformed 'create-qr-code' API request. Please consider the API
2 | documentation at http://goqr.me/api/doc/create-qr-code/
3
4
5 de: ungültige 'create-qr-code'-API-Anfrage. Bitte beachten Sie die
6 | API-Dokumentation unter http://goqr.me/de/api/doc/create-qr-code/
```

3. Salida

Tipo de dato	Observaciones
Respuesta de imagen	No es en JSON

API para validar número de DNI:

URL	Method
https://api.peruapis.com/v1/dni	GET

1. Argumento de entrada:

Campo	Tipo de dato	Observaciones
document	String	Número de DNI para obtener la información de la persona

- **Auth:**

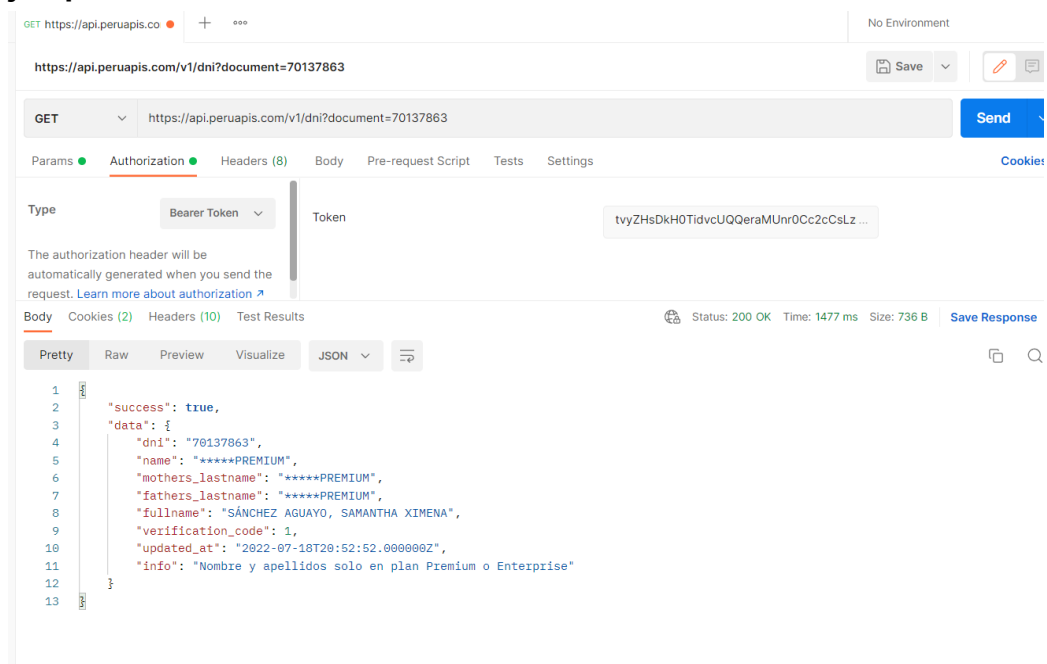
Bearer Token:

yOgeqH51FG63iLlp5CYyMyi3D0jaSEF9hDj3HS7IbCg20YJh1m65Fi4c8IyW

2. Argumento de salida

- **HTTP Status:** 200 OK

- **Ejemplo:**



3. Datos de argumentos de salida

Campo	Tipo de dato	Observaciones
dni	int	Número de 8 dígitos
name	String	Campo oculto

mothers_lastname	String	Campo oculto
fathers_lastname	String	Campo oculto
fullname	String	-
verification_code	int	-
updated_at	Datetime	-
info	String	-

- **HTTP Status:** 200 OK
- **Body:**

Observaciones: Si se le ingresa un valor diferente a los números de 8 dígitos, dará como respuesta lo siguiente:

The screenshot shows a REST client interface with the following details:

- URL:** `https://api.peruapis.com/v1/dni?document=701378633`
- Method:** GET
- Authorization:** Bearer Token with token `tvYzHsDkH0TidvcUQQeraMUnr0Cc2cCsLz ...`
- Status:** 200 OK, Time: 240 ms, Size: 382 B
- Response Body (JSON):**

```

1  {
2    "success": true,
3    "message": "API OK",
4    "status": "online"
5  }

```

API para verificar tarjeta de crédito:

URL	Method
http://apitarjeta.japaneast.cloudapp.azure.com/tarjeta/verificar	POST

1. Argumento de entrada:

Campo	Tipo de dato	Observaciones
numero	String	El número ingresado debe tener un total de 16 dígitos
fechaexpiracion	String	La fecha ingresada debe el formato mm/yyyy (mes/año).
cvv	String	El valor ingresado debe ser de 3 dígitos.

- **Body:**

```
{  
  "numero":"5420203273186412",  
  "fechaexpiracion":"07/2025",  
  "cvv":"478"  
}
```

- **Auth:**

Basic auth:

Username: teleticket

Password: 4675bedyv86g8nyiv21s07

2. Argumento de salida:

- **HTTP Status:** 200 OK

- **Body:**

```
{  
  "estado": "Verificado",  
  "informacion": {  
    "banco": "Banco Ripley",  
    "pais": "Perú",  
    "metodopago": "MasterCard"  
  }  
}
```

- Ejemplo:

The screenshot shows a REST client interface with the URL `http://apitarjeta.japaneast.cloudapp.azure.com/tarjeta/verificar`. The request is a POST with a JSON body:

```
1 {
2   "numero": "5420203273186412",
3   "fechaexpiracion": "07/2025",
4   "cvv": "478"
5 }
```

The response is shown in the 'Body' tab, displaying a JSON object with the following structure:

```
1 {
2   "estado": "Verificado",
3   "informacion": {
4     "banco": "Banco Ripley",
5     "pais": "Perú",
6     "metodopago": "MasterCard"
7   }
8 }
```

The status bar indicates a 200 OK response with a time of 675 ms and a size of 449 B.

En caso la tarjeta no sea válida:

The screenshot shows the same REST client interface with the same URL and request body as the previous example. The response is shown in the 'Body' tab, displaying a JSON object indicating an invalid card:

```
1 {
2   "msg": "No se ha encontrado la tarjeta de credito con los datos indicados",
3   "estado": "No existe"
4 }
```

The status bar indicates a 200 OK response with a time of 1146 ms and a size of 442 B.

3. Datos de argumentos de salida

Campo	Tipo de dato	Observaciones
estado	String	Estado de la tarjeta
informacion		
informacion.banco	String	Banco emisor de la tarjeta
informacion.pais	String	País del banco emisor
informacion.metodopago	String	Método de pago de tarjeta (Visa, MasterCard, etc)
msg	String	Mensaje en caso de error