# DocuScrape Extraction API — Request Parameters

Some dockets accept a **parameter** that tells DocuScrape which version of a document you are sending (for example the old or the new Uganda National ID card). You send the parameter and two generic files, `front` and `back`. You never need to know DocuScrape's internal folder names.

Base URL: `https://<your-docuscrape-host>` Auth: `Authorization: Bearer <your-api-key>`

## 1. See which parameters your docket accepts

```
GET /v1/dockets/{docketId}/parameters
```

```
{
  "docketId": "docket-910",
  "parameters": {
    "cardtype": { "values": { "old": { "slots": ["back", "front"] },
                              "new": { "slots": ["back", "front"] } } }
  }
}
```

`cardtype` is the parameter name, `old` / `new` are its allowed values, and `front` / `back` are the file parts you must send for that value.

## 2. Send a document

```
POST /v1/dockets/{docketId}/documents/extract        (multipart/form-data)
```

| Part | Required | Meaning |
|---|---|---|
| `documentId` | yes | Your own id for this document (returned to you unchanged) |
| `cardtype` | yes | `old` or `new` (case-insensitive) |
| `front` | yes | Image of the front of the card (JPEG/PNG) |
| `back` | yes | Image of the back of the card (JPEG/PNG) |

Headers: `Authorization: Bearer <key>`, and `Idempotency-Key: <unique value>` (recommended — retrying the same request with the same key never creates or charges a second time).

```
curl -X POST "https://<host>/v1/dockets/docket-910/documents/extract" \
  -H "Authorization: Bearer <your-api-key>" \
  -H "Idempotency-Key: 8f2c-unique-per-document" \
  -F documentId=your-id-123 \
  -F cardtype=new \
  -F front=@front.jpg \
  -F back=@back.jpg
```

Choose the value that matches the card in the photo (`old` for the previous layout, `new` for the current one). Your application decides this; DocuScrape does not guess it for you. Each side uses 1 credit.

## 3. Read the result

The call waits up to ~30 seconds. If `status` is `QUEUED` or `PROCESSING` (HTTP 202), poll `GET /v1/extractions/{requestId}` every few seconds until it finishes.

```
{
  "requestId": "dsreq-…",
  "documentId": "your-id-123",
  "status": "COMPLETED",
  "parameters": { "cardtype": "new" },
  "sides": {
    "front": { "status": "COMPLETED",
               "fields": [ { "name": "NIN", "value": "…", "confidence": 96.2, "status": "EXTRACTED" } ] },
    "back":  { "status": "COMPLETED", "fields": [ … ] }
  },
  "crossChecks": [ { "check": "NIN_MRZ_CHECKSUM", "status": "VERIFIED" } ]
}
```

- Results are under `sides.front` and `sides.back`.
- Overall `status`: `COMPLETED`, `COMPLETED_WITH_WARNINGS` (something needs a look), or `FAILED`.
- Field `status`: `EXTRACTED`, `LOW_CONFIDENCE` (treat as unverified), `NOT_FOUND`, `INVALID_FORMAT`.
- `crossChecks` compares the front with the machine-readable zone on the back. If a value was corrected by a checksum, the field has `correctedFrom` showing the original read.
- Field names differ slightly between versions (the new card adds `OTHER-NAME`, `DATE-OF-ISSUE` and `SERIAL-NO.`); `NIN`, `SurName`, `Given Name`, `DATE-OF-BIRTH` exist in both.

## Errors

Errors with a code look like `{"detail": {"error": {"code": "…", "message": "…"}}}`. A few (401, 402) return a plain `{"detail": "message"}`, so read the HTTP status first.

| HTTP | Code | Meaning |
|---|---|---|
| 400 | `PARAMETER_REQUIRED` | You did not send a parameter (the message lists the allowed ones) |
| 400 | `INVALID_PARAMETER_VALUE` | The value is not allowed (the message lists the allowed ones) |
| 400 | `MISSING_DOCUMENT` | A required file part (`front` or `back`) is missing |
| 400 | `UNEXPECTED_DOCUMENT` | A file part that this value does not use was sent |
| 401 | — | Missing or wrong API key |
| 404 | `DOCKET_NOT_FOUND` | Your key does not belong to this docket |
| 409 | `IDEMPOTENCY_CONFLICT` | This `Idempotency-Key` was already used with different files |
| 402 | — | Not enough credits for this request (1 per side) — top up and retry |

Keep your API key secret: send it only from your server, never from a browser or mobile app.