

# Block Chain: Bloxberg 2.0

- [\[1\], \[2\], \[3\], \[4\]](#)
- [\[5\]](#)
- [Example: Photo ID](#)
- [Example: Bardcodes](#)
- [Example Photo Id + multiple images and barcodes](#)

?, ?, ?, ?

A very simple and a very short but a very powerful program:

```
Ⓛ  
  
show (ⓁⓁ);  
show (ⓁⓁ);  
  
show ⓁⓁalso Ⓛ  
  
save as blox;
```

Output

"Ⓛ" → "0x40d93f50D4a56Be60fDe352c01097E1E42405b69"

"Ⓛ" → "0xa020807ff2c90743afbc8c5f74c191cb6eb037f1613211d42a4bf3977e5f9f0"

"Ⓛ" → <|"Type" -> "EllipticCurve", "CurveName" -> "secp256k1", "Compressed" -> False,

"PublicByteArray" -> {4, 250, 117, 78, 63, 249, 35, 37, 119, 220, 59, 73, 128, 250, 4, 178, 253, 14,  
126, 0, 19, 129, 44, 182, 137, 197, 161, 110, 203, 53, 24, 120, 64, 179, 9, 240, 194, 86, 37, 144,  
38, 84, 88, 224, 184, 173, 242, 231, 142, 52, 104, 62, 26, 166, 48, 141, 145, 95, 228, 62, 101, 132,  
235, 155, 110},

"PublicCurvePoint" ->

{11328547331332457675378811984661559764481072006219424404727484241178193248057  
6,  
80981563178412881780651944120220270359429660115039823663624727923423640787822}

|>

"Ⓛ" → <|"Type" -> "EllipticCurve", "CurveName" -> "secp256k1",

"PublicCurvePoint" ->

{11328547331332457675378811984661559764481072006219424404727484241178193248057  
6,  
80981563178412881780651944120220270359429660115039823663624727923423640787822},

"PrivateMultiplier" ->

72427481752409419219811996059199421334290305641324603076644566044143119628784,

"Compressed" -> False,

"PublicKey" -> {4, 250, 117, 78, 63, 249, 35, 37, 119, 220, 59, 73, 128, 250, 4, 178, 253, 14, 126, 0, 19, 129, 44, 182, 137, 197, 161, 110, 203, 53, 24, 120, 64, 179, 9, 240, 194, 86, 37, 144, 38, 84, 88, 224, 184, 173, 242, 231, 142, 52, 104, 62, 26, 166, 48, 141, 145, 95, 228, 62, 101, 132, 235, 155, 110},

"PrivateKey" -> {160, 32, 128, 127, 242, 201, 7, 67, 175, 188, 184, 197, 247, 76, 25, 28, 182, 235, 3, 127, 22, 19, 33, 29, 66, 164, 191, 57, 119, 229, 249, 240}

|>

??

```
␣␣  
  
␣␣ (  
␣␣"dara1"->"dara tag1"  
␣"this is a ccndev test1"  
␣␣"dara2"->"dara tag2"  
␣"this is a ccndev test2"  
␣);  
  
keys = "␣␣/bloxberg/keys";  
  
show keys;  
  
value = "␣␣/pdf";  
cert = "␣␣/certificate";  
crid = "␣␣/bloxberg/crid";  
bodycrid = "␣␣/hashlist";  
  
show value also cert also crid also bodycrid;  
  
save as blox2;
```

## Output

"keys" → "{@context, id, type, issuer, issuanceDate, credentialSubject, crid, cridType, metadataJson, proof}"

"value" →

"https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert\_pdf/ff\_free\_blox2.pdf"

"cert" →

"https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert\_json/ff\_free\_blox2.json"

"crid" → "9fc15cac82ea792549e70d7f7ed796b87933c5cae4785250939fc745b55c1cae"

"bodycrd" → "{9fc15cac82ea792549e70d7f7ed796b87933c5cae4785250939fc745b55c1cae,54b6e8fda68070499e08cf95d11fa56db20f5bda24d1237f4f9bcdcfaaecddee}"



ISSUES THIS  
**RESEARCH OBJECT CERTIFICATE**  
**#164**

The bloxberg certificate serves as a proof of existence that the data corresponding to the cryptographic identifier were transacted on the bloxberg blockchain at the issued time.

MINTED NFT	TRANSACTION DETAILS
<a href="https://explorer.l2.bloxberg.org/token/0x7B7BE38692557E4f033bd8bE0464cE7fc8F4C41/instance/164">https://explorer.l2.bloxberg.org/token/0x7B7BE38692557E4f033bd8bE0464cE7fc8F4C41/instance/164</a>	0x03dc3426261e6f1abd34a3a40a7a3e5df882a864f9546604a9bd170717dae154
FILE HASH	TIME STAMP
54b6e8fda68070499e08cf95d11fa56db20f5bda24d1237f4f9bcdcfaaecddee	2026-05-26T02:23:14.476510



```
"[
  {
    "@context": [
      "https://www.w3.org/2018/credentials/v1",
      "https://w3id.org/bloxberg/schema/research_object_certificate_v1"
    ],
    "id": "https://bloxberg.org",
    "type": [
      "VerifiableCredential",
      "BloxbergCredential"
    ],
    "issuer": "https://raw.githubusercontent.com/bloxberg-org/issuer_json_l2/master/issuer.json",
    "issuanceDate": "2026-05-26T00:49:31.743598+00:00",
    "credentialSubject": {
```

```
  \"id\": \"https:
//sepolia.arbiscan.io/address/0x3A2165222F8189A2Ed880e555868BeE1C67963e9\",
  \"issuingOrg\": {
    \"id\": \"https://bloxberg.org\"
  }
},
\"crid\": \"9fc15cac82ea792549e70d7f7ed796b87933c5cae4785250939fc745b55c1cae\",
\"cridType\": \"sha2-256\",
\"metadataJson\": \"{
  \\\"dara1\\\": \\\"daratag1\\\",
  \\\"dara2\\\": \\\"daratag2\\\"
}\",
\"proof\": {
  \"type\": \"MerkleProof2019\",
  \"created\": \"2026-05-26T00: 49: 32.325863\",
  \"proofValue\":
\"z9nmN4tsD8AYm9oXySiMhLVX9vwX5XjRDJiYU4fJv9r5DoZFnxWHXSfG3wutGpnjLCTvoKVF8X5YSyikB6rFAyoNkUY3
U6SMNF42WNToyFRYZPPHbvU1pPvWcRKggfTMmJAKFTPr3Ef9iFSpKp2kiRc17PKfxBUHUsQ6cc65Yt6NyC92Td3RbpyWNT
EvXP5Zt7dS1dXNd56P9fJs1THhB59iQUvL4zvkb5uTGHsHA3yk7M53oxorzFDZgvH5CNnrgPBBcwVNWnkBUiLprjwPowfo
7Se56buCJ5p37rr1mqaeEq8k9QYLAqKdB4hVpHEDAX1NWJQ75wa5ZDXvPegmpDxZ44PD9hokgnQGh8mh7f2rBgmnh9uqgv
pnjsgzAP1b3NxCP9C5yZZTWcNKfJmthUk4UuU\",
  \"proofPurpose\": \"assertionMethod\",
  \"verificationMethod\": \"ecdsa-koblitz-pubkey:
0x995b5A82500bc2558Fd37937e795f6468F14642c\",
  \"ens_name\": \"mpdl.berg\"
}
},
{
  \"@context\": [
    \"https://www.w3.org/2018/credentials/v1\",
    \"https://w3id.org/bloxberg/schema/research_object_certificate_v1\"
  ],
  \"id\": \"https://bloxberg.org\",
  \"type\": [
    \"VerifiableCredential\",
    \"BloxbergCredential\"
  ],
  \"issuer\": \"https://raw.githubusercontent.com/bloxberg-
org/issuer_json_l2/master/issuer.json\",
  \"issuanceDate\": \"2026-05-26T00: 49: 31.743598+00: 00\",
```

```
\ "credentialSubject\ ": {
  \ "id\ ": \ "https:
//sepolia.arbiscan.io/address/0x3A2165222F8189A2Ed880e555868BeE1C67963e9\ ",
  \ "issuingOrg\ ": {
    \ "id\ ": \ "https: //bloxberg.org\ "
  }
},
\ "crId\ ": \ "54b6e8fda68070499e08cf95d11fa56db20f5bda24d1237f4f9bcdfaaecddee\ ",
\ "crIdType\ ": \ "sha2-256\ ",
\ "metadataJson\ ": \ "{
  \ \ "dara1\ \ ": \ \ "daratag1\ \ ",
  \ \ "dara2\ \ ": \ \ "daratag2\ \ "
}\ ",
\ "proof\ ": {
  \ "type\ ": \ "MerkleProof2019\ ",
  \ "created\ ": \ "2026-05-26T00: 49: 32.329260\ ",
  \ "proofValue\ ":
\ "z9nmN4tsCzSUK5dJjPu71x2JjnpQCKMnVFzLXbTeEx89caacHd1rFHwKMfoAWJzTYzUHJurvYiwpdoTx8FjdBebr31Kv
9oNvhCrubyXCoA5yxu5fKYKqo6uTQgLqmTwmxE7vuUXhWYhDcSvuXTTEguyPa3PyfgdyuAqDLmQgXKhT7d7TCF2qs67Epg
effJLlkgE39z79smuQfmnAD9YWAhZ7XqayCxiabCLpG4jSDKgbdB8V4x2dAwCxf3V6xcsRf5GxbhnZzo9T2ATfYq74D67d
gpVSYPa8gWAtYWZst2EFKwLji2ZJZy2jBJvKg7uaFsypNpow3g4UXYed8C5S1vLykfCHviz1gRBH9ru8msWs9TMp2uYoPX
VcsJ68Hyx25SsxP4MSpYKCPFABNny6BBamefx\ ",
  \ "proofPurpose\ ": \ "assertionMethod\ ",
  \ "verificationMethod\ ": \ "ecdsa-koblitz-pubkey:
0x995b5A82500bc2558Fd37937e795f6468F14642c\ ",
  \ "ens_name\ ": \ "mpdl.berg\ "
}
}
]"
```

# Example: Photo ID

```
{}  
{}  
  
//add name and other information;  
{}(  
{}"first name"->"dara"  
{}"last name"->"shayda"  
);  
  
show "{}highlight";  
  
save as photo_id;
```



# Example: Bardcodes

Any image/photo entering the Free Form language computational container is immediately processed for:

- Any faces? if yes, highlight and cutout the facial regions and compute them (their properties).
- Any barcodes or qr-codes? if yes, highlight the barcode regions and read them.
- Any texts? if yes, OCR the image.
- Add the necessary post-processing e.g. image annotations
- Finally collect at the resulting metadata tags and contents (to be hashed)  $\varphi$  and URL export as String



```
//symbol to process barcodes and qrcodes;  
□□□□□□□□;□□□□□□  
  
□□  
  
//add name and other information;  
□□ (  
□□first name"->"dara"  
□□last name"->"shayda"  
);  
  
show "□□□□□□□□;□□□□□□";  
  
show "□□□□□□□□;□□□□□□";
```

```
save as blox_qr;
```

## Output

```
"[ ]/[index]"
```



```
"[ ]/[bartexts]"
```

```
{"065656915641", "065515000037", "9771468413022", "037000377085", "8435039118747"}
```

## Blockchain Certificate

```
"[ ] /certificate" (trimmed text)
```

[https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert\\_json/ff\\_free\\_blox\\_qr.json](https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert_json/ff_free_blox_qr.json)

There are currently 4 default mandatory items inside the hash list or the CRID and you can obtain the barcode related ones

Scanned bars as above

[https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff\\_free\\_blox\\_qr\\_bars.png](https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_qr_bars.png)

JSON containing if any other OCR texts computed from the image

[https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff\\_free\\_blox\\_qr\\_imagetext.json](https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_qr_imagetext.json)

See the bounding box JSON below addresses as "\_boxes"

[https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff\\_free\\_blox\\_qr\\_boxes.json](https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_qr_boxes.json)

The String values for the barcodes and the qrcodes

[https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff\\_free\\_blox\\_qr\\_barcodebox.json](https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_qr_barcodebox.json)

"□□□□□□□□□□/φ"

"φ"https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff\_free\_blox\_qr\_bars.png"φ"https:

## qr-codes or bar-codes bounding box scanning area

Any geometrical information pertinent to full understanding and computational access for a region of an image or a recognized object is computed and place in appropriate JSON structures.

"□□□□□□□□□□/boxes"

[

□

□

4.295e2,

7.305e2

□

□

8.395e2,

9.105e2

□

□

□

□

11.395e2,

13.925e2

□

□

7.745e2,

8.205e2

□

□

□

□

9.805e2,

8.085e2

□

□

9.855e2,

8.805e2

□

□

□

□

1.1775e3,

8.555e2

□

□

1.2515e3,

9.075e2

□

0

0

0

0.565e2,

0.285e2

0

0

0.365e2,

0.395e2

0

0

]

# Example Photo Id + multiple images and barcodes

- USA, the state of Virginia (proposed?) Personal Identity card
- With qr codes, barcodes and multiple images and text.
- Symbol `Ⓜ` does all the facial computing (neural nets), barcodes, qr codes and image text
- This will empower a universal blockchain or bloxberged Digital object that could and should enable a new de-centralized identification card system.

`Ⓜ`

`Ⓜ`

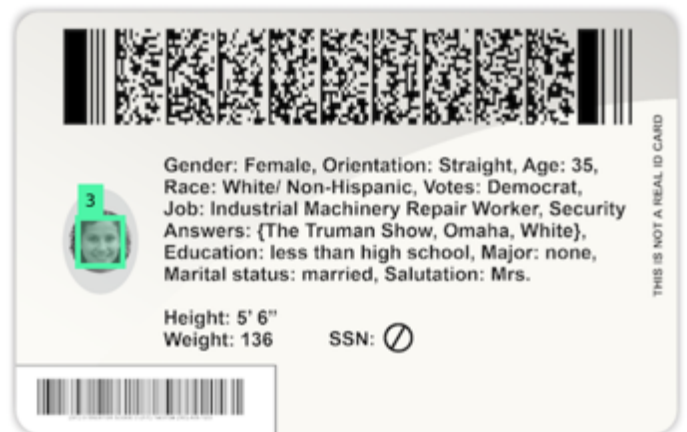
```
//Ⓜfills in numerous Ⓜcontent + Ⓜmetadata into the blockchain;
```

```
Ⓜ;
```

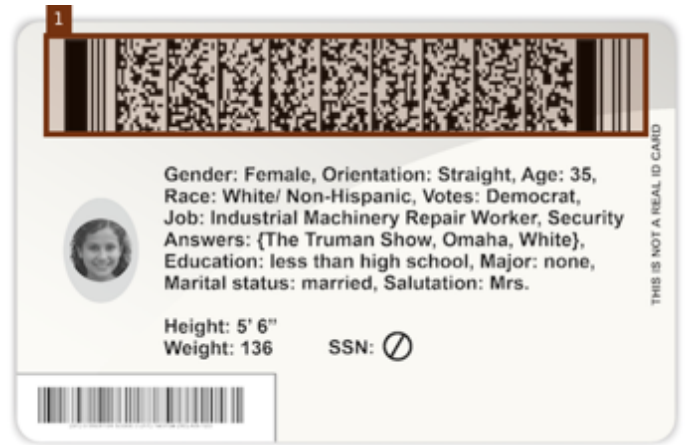
```
show "Ⓜindex";
```

```
save as photo_id;
```

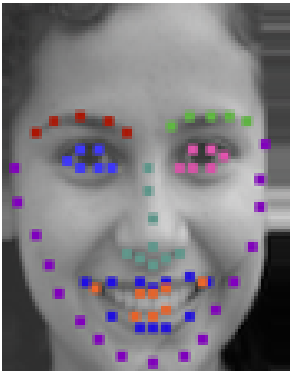
"Ⓜ/index"



"Ⓜ/barsindex"



"/highlight"



"/text"

```
{
  "image_text": "Vi soe VA, USA FEDERAL *LYQIM1A IDENTIFICATION CARD
  _#n* a716700487 eTESSIER 8 Gender: Female, Orientation: Straight,
  Age: 35, BELLA Fi Race: White/ Non-Hispanic, Votes: Democrat, Ad
  srw aD *B88795-a202 5 Job: Industrial Machinery Repair Worker,
  Security74 IN RD: 5 5 Answers: {The Truman Show, Omaha, White}, n
  Dato of birt ry Marital status: married, Salutation: Mrs. nOeio4rt988
  A\ na ED Height: 5 6  F BRO Weight: 136 ssn: QNONE nntek
  items tebe IMT |"
}
```

"/bartexts"

```
{
"@<LF><FS><CR>ANSI
6360050101DL00300203DLDAQ3265188<LF>DAALOTT,ERIC,B,<LF>DAG763 TEST
STREET<LF>DAINEW YORK CITY<LF>DAJSC<LF>DAK10005 <LF>DARD <LF>DAS
<LF>DAT <LF>DAU601<LF>DAW170<LF>DAYBRO <LF>DAZBRO
<LF>DBA20241004<LF>DBB19911004<LF>DBC1<LF>DBD20140101<LF>DBG2<LF>DBH1
}
```

## Color Keys

Blockchain by Bloxberg 2.0

CCN cloud systems

## Image Processing + Bloxberg 2.0 Blockchain Digitals

```
{
"/face", "/face1", "/highlight", "/faces", "/index", "/text",
"/facials", "/boxes", "/bartexts", "/barsindex", "/?", "/", "
processed", "( )", "/type", "/curve", "/compressed", "( )",
"( )", "( )/type", "/type", "( )", "( )", "( )/type", "/type",
"/bloxberg/contexts", "/bloxberg/context1/@context",
"/bloxberg/context1/id", "/bloxberg/context1/type",
"/bloxberg/context1/issuer", "/bloxberg/context1/issuancedate",
"/bloxberg/context1/credentialsubject", "/bloxberg/context1/crid",
"/bloxberg/context1/cridtype", "/bloxberg/context1/metadataajson",
"/bloxberg/context1/proof", "/bloxberg/context2/@context",
"/bloxberg/context2/id", "/bloxberg/context2/type",
"/bloxberg/context2/issuer", "/bloxberg/context2/issuancedate",
```

"□ /bloxberg/context2/credentialsubject", "□ /bloxberg/context2/crid",  
"□ /bloxberg/context2/cridtype", "□ /bloxberg/context2/metadajson",  
"□ /bloxberg/context2/proof", "□ /bloxberg/context3/@context",  
"□ /bloxberg/context3/id", "□ /bloxberg/context3/type",  
"□ /bloxberg/context3/issuer", "□ /bloxberg/context3/issuancedate",  
"□ /bloxberg/context3/credentialsubject", "□ /bloxberg/context3/crid",  
"□ /bloxberg/context3/cridtype", "□ /bloxberg/context3/metadajson",  
"□ /bloxberg/context3/proof", "□ /bloxberg/context4/@context",  
"□ /bloxberg/context4/id", "□ /bloxberg/context4/type",  
"□ /bloxberg/context4/issuer", "□ /bloxberg/context4/issuancedate",  
"□ /bloxberg/context4/credentialsubject", "□ /bloxberg/context4/crid",  
"□ /bloxberg/context4/cridtype", "□ /bloxberg/context4/metadajson",  
"□ /bloxberg/context4/proof", "□ /bloxberg/context5/@context",  
"□ /bloxberg/context5/id", "□ /bloxberg/context5/type",  
"□ /bloxberg/context5/issuer", "□ /bloxberg/context5/issuancedate",  
"□ /bloxberg/context5/credentialsubject", "□ /bloxberg/context5/crid",  
"□ /bloxberg/context5/cridtype", "□ /bloxberg/context5/metadajson",  
"□ /bloxberg/context5/proof", "□ /bloxberg/context6/@context",  
"□ /bloxberg/context6/id", "□ /bloxberg/context6/type",  
"□ /bloxberg/context6/issuer", "□ /bloxberg/context6/issuancedate",  
"□ /bloxberg/context6/credentialsubject", "□ /bloxberg/context6/crid",  
"□ /bloxberg/context6/cridtype", "□ /bloxberg/context6/metadajson",  
"□ /bloxberg/context6/proof", "□ /bloxberg/context7/@context",  
"□ /bloxberg/context7/id", "□ /bloxberg/context7/type",

```
" /bloxberg/context7/issuer", " /bloxberg/context7/issuancedate",  
" /bloxberg/context7/credentialsubject", " /bloxberg/context7/crid",  
" /bloxberg/context7/cridtype", " /bloxberg/context7/metadatatjson",  
" /bloxberg/context7/proof", " /contents", " /hashlist", " /key_eth",  
" /value", " /pdf", " /fullcredentialobject",  
" /certificate", " /metadata",  
}
```

```
" /"
```

```
"\https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_id_face.png"\  
https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_id_facehighlight.png"\  
https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_id_facehighlights.png"\  
https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_id_index.png"\  
https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_id_imagetext.json"\  
https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_id_facialpoints.json"\  
https://www.wolframcloud.com/obj/ccn2/freeform/ff/images/ff_free_blox_id_barcodebox.json"
```

```
" /"
```

```
"group"->"ff" "user"->"free" "ffname"->"blox_id" "publisher"->"CCN"  
"license"->"CC-BY-NC-ND 4.0"
```

## Full Credential Object

```
" /fullcredentialobject"
```

```
("{"@context":["https://www.w3.org/2018/credentials/v1","https://w  
3id.org/bloxberg/schema/research_object_certificate_  
v1"],"id":["https://bloxberg.org"],"type":["VerifiableCredential  
","BloxbergCredential"],"issuer":["https://raw.githubusercontent.c  
om/bloxberg-org/issuer_json_  
12/master/issuer.json"],"issuanceDate":["2026-06-02T02:59:59.
```

448151+00:00\", \"credentialSubject\": {\"id\": \"https://sepolia.arbiscan.io/address/0xAF3aD7B80e90C0Ca651a8e673E193eCC690BADA8\", \"issuingOrg\": {\"id\": \"https://bloxberg.org\"}}, \"cred\": \"9d05f7aa5009599ded40ff901de8fbce8bedb00004c0e6ad77fd128c7fac6806\", \"credType\": \"sha2-256\", \"metadata\": {\"group\": \"ff\", \"user\": \"free\", \"fname\": \"blox\_id\", \"publisher\": \"CCN\", \"license\": \"CC-BY-NC-ND 4.0\"}}, \"proof\": {\"type\": \"MerkleProof2019\", \"created\": \"2026-06-02T02:59:59.947487\", \"proofValue\": \"zLBHrm1GqmftAu35gTndQtHSErc2pdrx2dQ69dF9FqvjvqCY71iNnFEaW1kSw5rVcT6XXiWJHY7uUQyKhjPsg32KpHfyMR2GZZkEeY7YvcRRQSAjKksduAX2KXxWnFFBWFbn27m7m9ritdwzTzbbGFPRNfAqSeGhDKxTqLp4wYgqfAWAuFxDnG3nhyUDCAP1c9BohuH45o3f64tEnAH3LQmJkzGKKixRqz9TtypnELSJwzeMPTWZNPTihHhL2PA9N9YvMt3RtbjGaWzRAwebi7Tf2FwujSNmCX7Cygb13 ...\"}

...  
@context\": [\"https://www.w3.org/2018/credentials/v1\", \"https://w3id.org/bloxberg/schema/research\_object\_certificate\_v1\"], \"id\": \"https://bloxberg.org\", \"type\": [\"VerifiableCredential\", \"BloxbergCredential\"], \"issuer\": \"https://raw.githubusercontent.com/bloxberg-org/issuer\_json\_12/master/issuer.json\", \"issuanceDate\": \"2026-06-02T02:59:59.947487\", \"proof\": {\"type\": \"MerkleProof2019\", \"created\": \"2026-06-02T02:59:59.947487\", \"proofValue\": \"zLBHrm1GqmftAu35gTndQtHSErc2pdrx2dQ69dF9FqvjvqCY71iNnFEaW1kSw5rVcT6XXiWJHY7uUQyKhjPsg32KpHfyMR2GZZkEeY7YvcRRQSAjKksduAX2KXxWnFFBWFbn27m7m9ritdwzTzbbGFPRNfAqSeGhDKxTqLp4wYgqfAWAuFxDnG3nhyUDCAP1c9BohuH45o3f64tEnAH3LQmJkzGKKixRqz9TtypnELSJwzeMPTWZNPTihHhL2PA9N9YvMt3RtbjGaWzRAwebi7Tf2FwujSNmCX7Cygb13 ...\"}

...  
@context\": [\"https://www.w3.org/2018/credentials/v1\", \"https://w3id.org/bloxberg/schema/research\_object\_certificate\_v1\"], \"id\": \"https://bloxberg.org\", \"type\": [\"VerifiableCredential\", \"BloxbergCredential\"], \"issuer\": \"https://raw.githubusercontent.com/bloxberg-org/issuer\_json\_12/master/issuer.json\", \"issuanceDate\": \"2026-06-02T02:59:59.947487\", \"proof\": {\"type\": \"MerkleProof2019\", \"created\": \"2026-06-02T02:59:59.947487\", \"proofValue\": \"zLBHrm1GqmftAu35gTndQtHSErc2pdrx2dQ69dF9FqvjvqCY71iNnFEaW1kSw5rVcT6XXiWJHY7uUQyKhjPsg32KpHfyMR2GZZkEeY7YvcRRQSAjKksduAX2KXxWnFFBWFbn27m7m9ritdwzTzbbGFPRNfAqSeGhDKxTqLp4wYgqfAWAuFxDnG3nhyUDCAP1c9BohuH45o3f64tEnAH3LQmJkzGKKixRqz9TtypnELSJwzeMPTWZNPTihHhL2PA9N9YvMt3RtbjGaWzRAwebi7Tf2FwujSNmCX7Cygb13 ...\"}

```
\":{\\"id\\":\\"https://bloxberg.org\\"}},\\"crid\\":\\"5b57331ee65751d0024a3
f47940cdecdeebf27e070c670cdee7e472aaaa274f3\\",\\"cridType\\":\\"sha2-256\\",\\"m
etadatajson\\":{\\"{\\\\"group\\\\":\\\\"ff\\\\", \\\\"user\\\\":\\\\"free\\\\"},
\\\\"ffname\\\\":\\\\"blox_id\\\\", \\\\"publisher\\\\":\\\\"CCN\\\\"},
\\\\"license\\\\":\\\\"CC-BY-NC-ND
4.0\\\\"}\\",\\"proof\\":{\\"type\\":\\"MerkleProof2019\\",\\"created\\":\\"2026-0
6-02T02:59:59.\\
951710\\",\\"proofValue\\":\\"zLBHrm1GqUpCCwm ....
...
"/pdf"
("{"https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert_pdf/ff_free_blox
_id_1.pdf",
\\"https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert_pdf/ff_free_blox_id
_2.pdf",
\\"https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxber2/cert_pdf/ff_free_blox_id_
3.pdf",
\\"https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert_pdf/ff_free_blox_id
_4.pdf",
\\"https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert_pdf/ff_free_blox_id
_5.pdf",
\\"https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert_pdf/ff_free_blox_id
_6.pdf",
\\"https://www.wolframcloud.com/obj/ccn2/freeform/ff/blockchain/bloxberg2/cert_pdf/ff_free_blox_id
_7.pdf"}})"
```

## Certificate 1



ISSUES THIS  
**RESEARCH OBJECT CERTIFICATE**  
**#206**

The bloxberg certificate serves as a proof of existence that the data corresponding to the cryptographic identifier were transacted on the bloxberg blockchain at the issued time.

MINTED NFT

<https://explorer.l2.bloxberg.org/token/0x787BE38692557E4f033bd8bE0464cE7fc8F4C41/instance/206>

TRANSACTION DETAILS

0xac074488c5a9fe1693fdaaca828e35f550245b21279edd62cc0f23fc7ea4981f

FILE HASH

00b073dd1b5c4711a5ac5d786c62b7076c6c32ee4822e1d049f72559eae72a7e

TIME STAMP

2026-06-02T02:59:59.959174



**Certificate 7**



ISSUES THIS  
**RESEARCH OBJECT CERTIFICATE**  
**#206**

The bloxberg certificate serves as a proof of existence that the data corresponding to the cryptographic identifier were transacted on the bloxberg blockchain at the issued time.

MINTED NFT

<https://explorer.l2.bloxberg.org/token/0x787BE38692557E4f033bd8bE0464cE7fc8F4C41/instance/206>

TRANSACTION DETAILS

0xac074488c5a9fe1693fdaaca828e35f550245b21279edd62cc0f23fc7ea4981f

FILE HASH

09789ec41c619eb8bca56c1ec3f89910917dfde6fa4da1c86dc50cdd5125a31f

TIME STAMP

2026-06-02T02:59:59.955489



