The Genesis Block

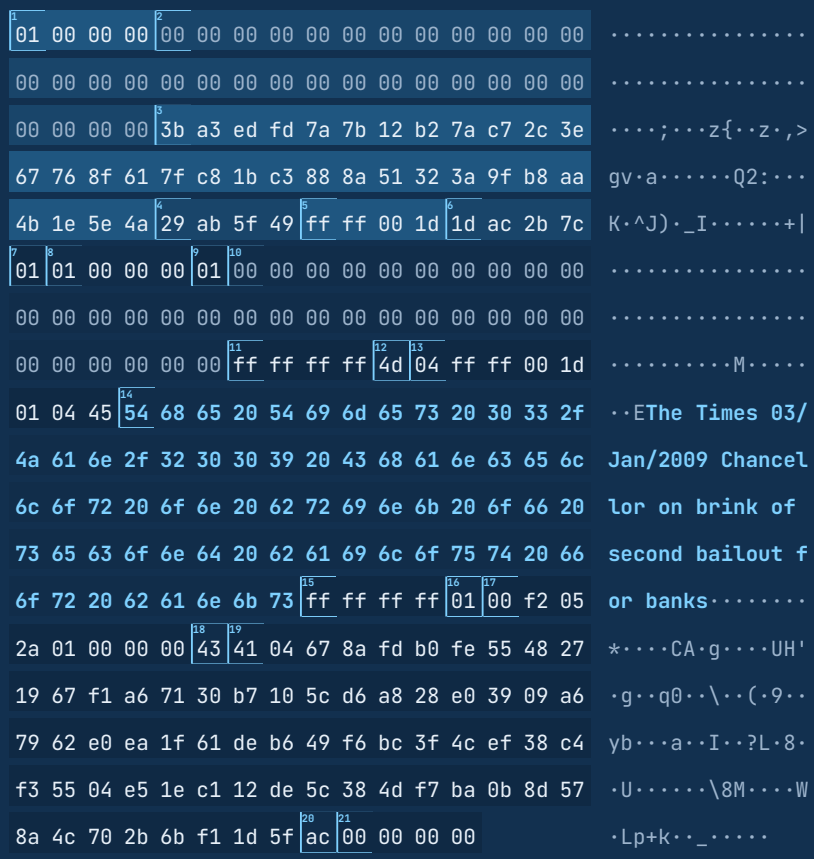
Bitcoin block 0, byte by byte

From the code of Satoshi Nakamoto and the Bitcoin Core developers

An annotated layout of the raw block, rebuilt from the source code of Bitcoin Core

0

block header, 80 bytes · the coinbase transaction, 205 bytes (after the count) · the message



The block header

1 **Version** offset 0, 4 bytes
`01000000`: 1, a signed 32-bit integer in little-endian order, as are all the numbers of the block.

2 **Previous block hash** offset 4, 32 bytes
32 zero bytes: there is no block before it. The source sets it with `hashPrevBlock.SetNull()`.

3 **Merkle root** offset 36, 32 bytes
The double SHA-256 of the only transaction, bytes 81 to 284. Shown, like every hash, in reverse byte order: `4a5e1e4b aab89f3a 32518a88 c31bc87f 618f7667 3e2cc77a b2127b7a fdeda33b`.

4 **Time** offset 68, 4 bytes
`29ab5f49`: 1231006505 seconds after 1 January 1970, that is 3 January 2009, 18:15:05 UTC.

5 **Bits** offset 72, 4 bytes
`ffff001d`: `0x1d00ffff`, the target in compact form: `0x00ffff` times 256^{26}, so a hash must start with 32 zero bits at least. It lies just under the proof-of-work limit of the source, `00000000ffff...ffff`, and Bitcoin Core rates it difficulty 1 (`GetDifficulty`, `src/rpc/blockchain.cpp`).

6 **Nonce** offset 76, 4 bytes
`1dac2b7c`: 2083236893, the number that made the hash of the header fall under the target.

The hash

SHA-256(SHA-256(bytes 0 to 79)) =
`6fe28c0a b6f1b372 c1a6a246 ae63f74f 931e8365 e15a089c 68d61900 00000000`

read in reverse byte order, the hash of block 0:
`00000000 0019d668 9c085ae1 65831e93 4ff763ae 46a2a6c1 72b3f1b6 0a8ce26f`

under the target of field 5:
`00000000 ffff0000 00000000 00000000 00000000 00000000 00000000 00000000`

The coinbase transaction

7 **Transaction count** offset 80, 1 byte
`01`: one transaction, a variable-length integer.

8 **Transaction version** offset 81, 4 bytes
`01000000`: 1.

9 **Input count** offset 85, 1 byte
`01`: one input.

10 **Previous output hash** offset 86, 32 bytes
32 zero bytes: a coinbase input spends no earlier output.

11 **Previous output index** offset 118, 4 bytes
`ffffffff`: the index −1 (4294967295) that marks a coinbase.

12 **Script length** offset 122, 1 byte
`4d`: 77 bytes of script follow.

13 **Coinbase script** offset 123, 8 bytes
`04 ff ff 00 1d 01 04 45`: `04`, push 4 bytes, `ffff001d`, the number 486604799 of the source, equal to the Bits of field 5; `01`, push 1 byte, `04`, the `CScriptNum(4)` of the source; then `45`, push 69 bytes.

The message

The Times 03/Jan/2009 Chancellor on brink of second bailout for banks

14 **Message** offset 131, 69 bytes
69 bytes of ASCII text, the `pszTimestamp` of the source.

15 **Sequence** offset 200, 4 bytes
`ffffffff`: the final sequence number.

16 **Output count** offset 204, 1 byte
`01`: one output.

17 **Value** offset 205, 8 bytes
`00f2052a01000000`: 5,000,000,000 satoshis, 50 bitcoins (`50 * COIN`, `COIN` being 100,000,000), a 64-bit integer.

18 **Script length** offset 213, 1 byte
`43`: 67 bytes of script follow.

19 **Public key** offset 214, 66 bytes
`41`, push 65 bytes: an uncompressed public key of the secp256k1 curve, `04` then its x and y coordinates.

20 **OP_CHECKSIG** offset 280, 1 byte
`ac`: `OP_CHECKSIG`. The 50 bitcoins go to whoever can sign for that key; the source notes that this output "cannot be spent since it did not originally exist in the database".

21 **Lock time** offset 281, 4 bytes
`00000000`: no lock time.

The public key

x = `678afdb0 fe554827 1967f1a6 7130b710 5cd6a828 e03909a6 7962e0ea 1f61deb6`

y = `49f6bc3f 4cef38c4 f35504e5 1ec112de 5c384df7 ba0b8d57 8a4c702b 6bf11d5f`

Sources

Every byte was rebuilt by a script from the values of `CreateGenesisBlock` in `src/kernel/chainparams.cpp` of Bitcoin Core (commit `ed7dd7c`, 25 September 2026, MIT license): the time 1231006505, nonce 2083236893, bits `0x1d00ffff`, version 1, reward `50 * COIN`, the text and the public key. The result hashes to the block hash and merkle root that the source asserts, and it is identical, byte for byte, to the raw block served by blockstream.info and by mempool.space. SHA-256 of the 285 bytes:
`5299fac924b5a2fc19a88876a0042c19ac4d11fe69c3f66e47516e26185f9e99`.

The MIT License (MIT) Copyright (c) 2009-2026 The Bitcoin Core developers Copyright (c) 2009-2026 Bitcoin Developers Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions: The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software. THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.