Genesis block Explanation

mixhash A 256-bit hash which proves, combined with the nonce, that a sufficient amount of computation has been carried out on this block: the Proof-of-Work (PoW). The combination of nonce and mixhash must satisfy a mathematical condition described in the Yellowpaper, 4.3.4. Block Header Validity, (44). It allows to verify that the Block has really been cryptographically mined, thus, from this aspect, is valid.
nonce A 64-bit hash, which proves, combined with the mix-hash, that a sufficient amount of computation has been carried out on this block: the Proof-of-Work (PoW). The combination of nonce and mixhash must satisfy a mathematical condition described in the Yellowpaper, 4.3.4. Block Header Validity, (44), and allows to verify that the Block has really been cryptographically mined and thus, from this aspect, is valid. The nonce is the cryptographically secure mining proof-of-work that proves beyond reasonable doubt that a particular amount of computation has been expended in the determination of this token value. (Yellowpager, 11.5. Mining Proof-of-Work).

difficulty A scalar value corresponding to the difficulty level applied during the nonce discovering of this block. It defines the mining Target, which can be calculated from the previous block's difficulty level and the timestamp. The higher the difficulty, the statistically more calculations a Miner must perform to discover a valid block. This value is used to control the Block generation time of a Blockchain, keeping the Block generation frequency within a target range. On the test network, we keep this value low to avoid waiting during tests, since the discovery of a valid Block is required to execute a transaction on the Blockchain.
alloc Allows defining a list of pre-filled wallets. That's an Ethereum specific functionality to handle the "Ether pre-sale" period. Since we can mine local Ether quickly, we don't use this option.

coinbase The 160-bit address to which all rewards (in Ether) collected from the successful mining of this block have been transferred. They are a sum of the mining reward itself and the Contract transaction execution refunds. Often named "beneficiary" in the specifications, sometimes "etherbase" in the online documentation. This can be anything in the Genesis Block since the value is set by the setting of the Miner when a new Block is created.

timestamp A scalar value equal to the reasonable output of Unix time() function at this block inception. This mechanism enforces a homeostasis in terms of the time between blocks. A smaller period between the last two blocks results in an increase in the difficulty level and thus additional computation required to find the next valid block. If the period is too large, the difficulty, and expected time to the next block, is reduced. The timestamp also allows verifying the order of block within the chain (Yellowpaper, 4.3.4. (43)).

parentHash The Keccak 256-bit hash of the entire parent block header (including its nonce and mixhash). Pointer to the parent block, thus effectively building the chain of blocks. In the case of the Genesis block, and only in this case, it's 0.

extraData An optional free, but max. 32-byte long space to conserve smart things for ethernity. :)

gasLimit A scalar value equal to the current chain-wide limit of Gas expenditure per block. High in our case to avoid being limited by this threshold during tests. Note: this does not indicate that we should not pay attention to the Gas consumption of our Contracts.