

HTTPH\$ - Set HTTP Server Request Header

The HTTPH\$ routine is used to establish a Request Header.

1. Invocation

To establish a Request Header code:

```
CALL HTTPH$ USING hh
```

where *hh* is a control block of the following format:

```
01  HH
02  HHVERS      PIC 9(4) COMP      * BLOCK VERSION NUMBER
                                     * MUST BE 1 or 2
```

*
*
*

If HHVERS = 1 the rest of the HH block is defined as follows:

```
02  HHHDL      PIC 9(4) COMP      * HTTP session handle
02  HHNAME     PIC X(100)         * Header name
02  HHVAL      PIC X(100)         * header value
```

*
*
*
*

For GSM SP-36, and later, if HHVERS = 2 the rest of the HH block is defined as follows:

```
02  HHHDL      PIC 9(4) COMP      * HTTP session handle
02  HHPNAME    PIC PTR           * Pointer to zero-terminated header name
02  HHPVAL     PIC PTR           * Pointer to zero-terminated header value
```

2. STOP Codes and Exception Conditions

The following STOP codes may be generated by HTTPH\$:

STOP code	Description
13612	HTTPH\$ has been called by an application that is not running on GX.
25022	An attempt has been made to call HTTPH\$ on an incompatible version of GX. The version of GX must be V3.5x, or later
13613	Invalid Control Block version
13614	The total length of the GX command block has exceeded an internal limit.

The following EXIT codes may be returned by HTTPH\$:

EXIT code	\$\$COND	Description
-----------	----------	-------------

13610	10	Unable to allocate memory for temporary work buffer
13611	11	The set HTTP server request header operation has suffered an exception. The error code has been returned in \$\$CRES.

3. Programming Notes

HTTPH\$ is only available when running on GX. Any attempt to use HTTPH\$ on a non-GX terminal will result in a STOP code. The version of GX must be V3.5x or later. The version of GSM must be GSM SP-17, or later. The version-2 control block (i.e. HHVERS=2) is only supported by GSM SP-36, and later.

The session handle returned by a previous, successful call to HTTPO\$ must be passed, via HHHDL, to HTTPH\$.

The Header Name (HHNAME) and Header Value (HHVAL) are normal character strings (i.e. fixed length with trailing SPACE characters). However, they are converted to zero terminated strings by the sub-routine that passes the strings to GX.

4. Examples

```
MOVE HTTPHDL TO HHHDL
MOVE "Content-Type" TO HHNAME
MOVE "text/xml" TO HHVAL
CALL HTTPH$ USING HH
```

5. Copy-Books

None.

6. See Also

HTTPC\$ Close HTTP server session
HTTPO\$ Open HTTP server session
HTTPSS\$ Send message and return status information and response text