

บทที่ 1

บทนำ

ในช่วงแรกๆ ภาษาที่นิยมใช้ในการพัฒนาเว็บไซต์ (Web Site) คือ ภาษา HTML (Hypertext Markup Language) โดยภายในหน้าเว็บเพจ (Web page) จะประกอบไปด้วย ตัวอักษร ภาพ หรืออื่นๆ ที่ไม่สามารถเปลี่ยนแปลงข้อมูลได้ด้วยตัวเอง ไม่สามารถคำนวณ ประมวลผลข้อมูล หรือเชื่อมต่อกับฐานข้อมูลได้ เป็นแค่ส่วนของการแสดงผลของหน้าเว็บเพจเท่านั้น หรือเรียกว่า หน้าเว็บเพจแบบคงที่ (Static Web page) ต่อมาได้พัฒนาภาษาที่ข้อมูลจะถูกเปลี่ยนแปลงโดยอัตโนมัติตามเงื่อนไขต่างๆ ที่ผู้เขียนกำหนดไว้ หรือเรียกว่า ไดนามิกเว็บเพจ (Dynamic Web page) ในปัจจุบันมีหลายภาษาที่ถูกพัฒนาขึ้นมา มีลักษณะเป็นภาษาสคริปต์ (Script Language) ที่สามารถติดต่อกับผู้ใช้ ฐานข้อมูล คำนวณ และประมวลผลตามเงื่อนไขต่างๆ ได้ และหนึ่งในภาษาสคริปต์ที่ได้รับความนิยมในปัจจุบันก็คือ ภาษา PHP (PHP Hypertext Preprocessor)

1.1 ประเภทของภาษาสคริปต์

ภาษาสคริปต์ มี 2 ประเภท ประกอบด้วย

1.1.1 ไคลเอนต์ไซด์สคริปต์ (Client Side script)

ไคลเอนต์ไซด์สคริปต์ คือ เทคโนโลยีที่สคริปต์มีการประมวลผลและแปลคำสั่ง (Statement Translation) ในฝั่งของเครื่องผู้ใช้ เช่น จาวาสคริปต์ ส่วนมากจะใช้วิธีการเขียนสคริปต์แทรกลงไปภาษา HTML และถูกประมวลผลโดยเว็บเบราว์เซอร์ (Web Browser) เพื่อโต้ตอบกับผู้ใช้ งาน เช่น การแสดงข้อความเตือน การตรวจสอบข้อมูลที่ผู้ใช้ป้อน เป็นต้น

1.1.2 เซิร์ฟเวอร์ไซด์สคริปต์ (Server Side script)

เซิร์ฟเวอร์ไซด์สคริปต์ คือ เทคโนโลยีที่สคริปต์ที่มีการประมวลผลและแปลคำสั่งในฝั่งของเครื่องเซิร์ฟเวอร์ (Server) แล้วแปลงเป็นเอกสารในรูปแบบของภาษา HTML แล้วส่งผลลัพธ์ที่ได้กลับไปให้เว็บเบราว์เซอร์ในฝั่งไคลเอนต์ (Client)

1.2 ความสามารถของภาษา PHP

PHP เป็นภาษาสคริปต์ที่มีการประมวลผลและแปลคำสั่งประเภทเซิร์ฟเวอร์ไซด์สคริปต์ มีรูปแบบในการเขียนคำสั่ง (Statement) การทำงานคล้ายกับภาษา Perl หรือภาษา C และสามารถทำงานร่วมกับภาษา HTML ได้เป็นอย่างดี ความสามารถของภาษา PHP มีรายละเอียด ดังต่อไปนี้

1.2.1 เป็นตัวแปลภาษาที่มีลักษณะ ที่เรียกว่า โอเพนซอร์ส (Open source) คือ ผู้ใช้สามารถดาวน์โหลดและติดตั้งบนเว็บเซิร์ฟเวอร์ (Web Server) ได้โดยไม่เสียค่าใช้จ่าย

1.2.2 เป็นภาษาสคริปต์ประเภทเซิร์ฟเวอร์ไซด์สคริปต์ มีการประมวลผลและแปลคำสั่งในฝั่งของเว็บเซิร์ฟเวอร์ แล้วแปลงเป็นเอกสารในรูปแบบของภาษา HTML แล้วส่งผลลัพธ์ที่ได้กลับไปให้เว็บเบราว์เซอร์ในฝั่งไคลเอนต์

1.2.3 สามารถทำงานบนระบบปฏิบัติการ (Operating System) ที่หลากหลาย เช่น Unix, MS-Windows, Mac OS หรือ Risc OS อย่างมีประสิทธิภาพ

1.2.4 สามารถทำงานร่วมกับเว็บเซิร์ฟเวอร์ เช่น Personal Web Server (PWS), Apache, OmniHttpd และ Internet Information Service (IIS) เป็นต้น

1.2.5 สามารถเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming) ได้

1.2.6 สามารถทำงานร่วมกับโปรแกรมระบบจัดการฐานข้อมูลที่หลากหลาย เช่น Oracle, MySQL, FilePro, Solid, FrontBase, mSQL และ MS SQL Server เป็นต้น

1.2.7 สามารถติดต่อและทำงานร่วมกับโปรโตคอล (Protocol) ชนิดต่างๆ ได้ เช่น LDAP, IMAP, SNMP, POP3 และ HTTP (Hypertext Transfer Protocol) เป็นต้น

1.3 ประวัติความเป็นมาของภาษา PHP



Rasmus Lerdorf



Zeev Suraski



Andi Gutmans

ภาพที่ 1.1 ผู้ร่วมพัฒนา PHP

ที่มา: Rasmus. (2011)., Markbuchner. (2010). และ Peter Judge. (2010).

ภาษา PHP ถูกสร้างขึ้นในปี ค.ศ.1994 โดย Rasmus Lerdorf เป็นผู้เริ่มต้นพัฒนา จุดเริ่มต้นมาจากความต้องการที่จะบันทึกข้อมูลผู้ที่เข้ามาเยี่ยมชมเว็บไซต์ส่วนตัวที่สร้างขึ้น โดยแนวคิดคือ การเขียนโปรแกรมด้วยภาษา C แต่ต้องการแยกส่วนที่เป็นภาษา HTML ออกจากภาษา C และเป็นเหตุผลสำคัญทำให้สร้างโค้ด (Code) ของภาษา HTML ขึ้นมาใหม่ และตั้งชื่อว่า Personal Home Page (PHP-Tools)

หลังจากพัฒนาภาษา PHP ขึ้นมาแล้ว ก็เริ่มแจกจ่ายให้บุคคลทั่วไปใช้งาน แต่ในช่วงแรกภาษา PHP ยังไม่มีความสามารถมากนัก ต่อมาในปี ค.ศ.1995 มีการพัฒนาเพิ่มขีดความสามารถให้กับภาษา PHP ให้สามารถรับข้อมูลที่ส่งมาจากฟอร์ม (Form) ของภาษา HTML รวมทั้งสามารถติดต่อกับฐานข้อมูล MySQL ได้อีกด้วย

ในปี ค.ศ.1997 ได้เปิดให้ผู้สนใจเข้าร่วมพัฒนาภาษา PHP และได้ผู้ร่วมพัฒนาเพิ่มอีก 2 คน คือ Zeev Suraski และ Andi Gutmans ได้ช่วยกันปรับปรุงและพัฒนาโค้ดขึ้นมาใหม่ให้ดีขึ้นในหลายๆ ด้าน ทั้งแก้ไขข้อบกพร่อง เพิ่มประสิทธิภาพ และเพิ่มฟังก์ชัน (Function) ใหม่ ๆ มากขึ้น และสามารถเขียนโปรแกรมเชิงวัตถุที่สมบูรณ์แบบ สามารถทำงานร่วมกับเว็บเซิร์ฟเวอร์ บนระบบปฏิบัติการหลายชนิด จนเป็นที่นิยมในปัจจุบัน และเปลี่ยนชื่อใหม่จากเดิม Personal Home Page (PHP-Tools) เป็น PHP Hypertext Preprocessor

ปัจจุบันภาษา PHP ได้ผ่านการพัฒนามาหลายเวอร์ชัน (Version) มีเว็บไซต์เพื่อให้ผู้ใช้หรือผู้พัฒนาเว็บไซต์สามารถเข้าไปดาวน์โหลดโปรแกรมเวอร์ชันล่าสุด รวมทั้งค้นหาข้อมูล วิธีการใช้งาน ฟังก์ชัน และสนทนากันระหว่างผู้ใช้ด้วยกัน เว็บไซต์ตั้งอยู่ที่ <http://php.net>

1.4 หลักการทำงานของภาษา PHP

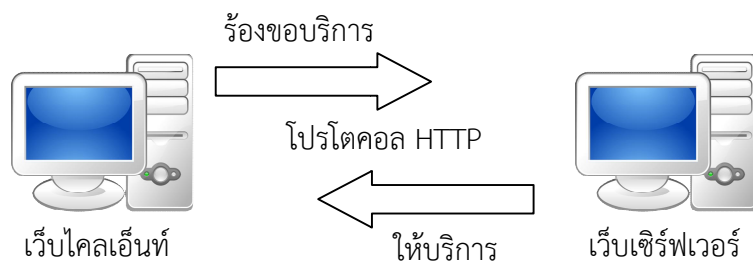
ภาษา PHP นั้น ทำงานอยู่ในฝั่งเว็บเซิร์ฟเวอร์ จะไม่สามารถเห็นสคริปต์ของภาษา PHP ที่เครื่องคอมพิวเตอร์ในฝั่งไคลเอ็นท์ เพราะระหว่างที่ใช้งานเว็บเบราว์เซอร์เรียกชมเว็บไซต์นั้น ในฝั่งของเว็บเซิร์ฟเวอร์ จะทำการประมวลผลภาษา PHP และส่งผลลัพธ์ที่ได้ เป็นภาษา HTML หลังจากนั้นจะส่ง HTML ที่ได้จากการประมวลผลแล้วนั้นมายังเว็บเบราว์เซอร์ของไคลเอ็นท์ แล้วแสดงผลให้เห็นเป็นรูปร่างของเว็บไซต์ และเมื่อดูซอร์สโค้ด (Source Code) ในฝั่งไคลเอ็นท์ จะเห็นเฉพาะ HTML tags ของภาษา HTML เพียงอย่างเดียว ไม่เห็นสคริปต์ PHP สามารถสรุปขั้นตอนและหลักการทำงานระหว่างไคลเอ็นท์ และเว็บเซิร์ฟเวอร์ ได้ดังต่อไปนี้

1.4.1 เว็บเซิร์ฟเวอร์ คือ โปรแกรมที่อยู่และทำงานบนเครื่องฝั่งเซิร์ฟเวอร์ ทำหน้าที่ในการรับคำสั่งจากการร้องขอของไคลเอ็นท์ (โดยผู้ใช้จะเรียกชมหน้าเว็บเพจได้โดยใช้โปรโตคอล HTTP ผ่านทางเว็บเบราว์เซอร์) และประมวลผลการทำงานจากการร้องขอดังกล่าว แล้วส่งข้อมูลกลับไปยังเครื่องของไคลเอ็นท์ที่ร้องขอ โดยสรุปคือ เว็บเซิร์ฟเวอร์จะคอยให้บริการแก่ไคลเอ็นท์ ที่ร้องขอข้อมูลเข้ามาโดยผ่านเว็บเบราว์เซอร์ หรือ จากการร้องขอผ่านโปรโตคอล HTTP โดยเว็บเพจที่เขียนก็สามารถเขียนได้หลายภาษา หลายรูปแบบ การเขียนสคริปต์ PHP ระบบจะสามารถทำงานได้นั้นจำเป็นต้องมีเว็บเซิร์ฟเวอร์ เป็นตัวเรียกใช้งาน Engine ของภาษา PHP ดังนั้นถ้าต้องการให้คอมพิวเตอร์สามารถประมวลผลและแปลภาษา PHP ได้นั้น จำเป็นต้องมีเว็บเซิร์ฟเวอร์หรือกรณีศึกษาที่สามารถจำลองเครื่องคอมพิวเตอร์ธรรมดาให้ทำหน้าที่เป็นเว็บเซิร์ฟเวอร์ (วิธีการติดตั้งสามารถศึกษาเพิ่มเติมได้ในภาคผนวก) รวมถึงรูปแบบ



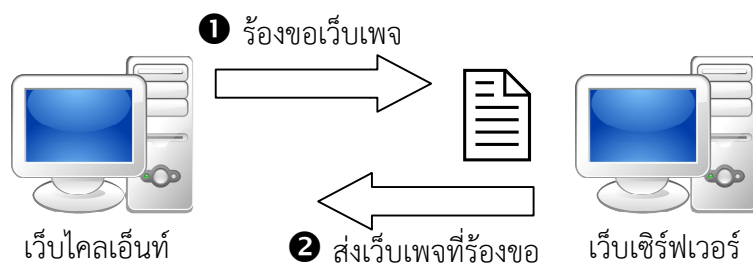
และวิธีการ เข้าถึงข้อมูล จะเรียกใช้ผ่านบริการจากโปรโตคอล HTTP เพื่อให้ผู้ใช้สามารถอ่านข้อมูล ภาพ เสียง หรือข้อมูลอื่นๆ ตามต้องการ

1.4.2 หลักการทำงานของเว็บเซิร์ฟเวอร์ มีลักษณะเช่นเดียวกับกับบริการอื่นๆ บนระบบ อินเทอร์เน็ต (Internet) คือ อยู่ในรูปแบบไคลเอ็นท์-เซิร์ฟเวอร์ (Client-Server) โดยมีโปรแกรมเว็บไคลเอ็นท์ (Web Client) ทำหน้าที่เป็นผู้ร้องขอบริการ และมีโปรแกรมเว็บเซิร์ฟเวอร์ ทำหน้าที่เป็นผู้ให้บริการ โปรแกรมเว็บไคลเอ็นท์ ก็คือโปรแกรมเว็บเบราว์เซอร์ในเครื่องของผู้ใช้ สำหรับโปรแกรมเว็บเซิร์ฟเวอร์นั้นจะถูกติดตั้งไว้ในเครื่องของผู้ให้บริการเว็บไซต์ (มักเรียกเครื่องคอมพิวเตอร์ที่ติดตั้งโปรแกรมเว็บเซิร์ฟเวอร์เพื่อให้บริการว่า "เว็บเซิร์ฟเวอร์") การติดต่อระหว่างโปรแกรมเว็บเบราว์เซอร์กับโปรแกรมเว็บเซิร์ฟเวอร์จะกระทำผ่านโปรโตคอล HTTP ดังภาพที่ 1.2



ภาพที่ 1.2 รูปแบบการติดต่อสื่อสารระหว่างเว็บไคลเอ็นท์และเว็บเซิร์ฟเวอร์ผ่านโปรโตคอล HTTP

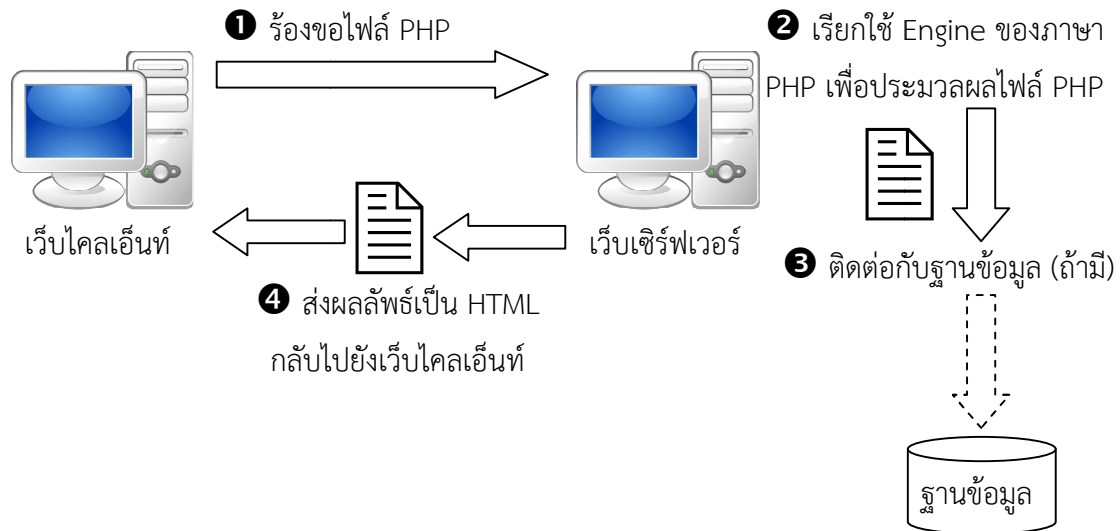
1.4.3 กลไกการทำงานเมื่อร้องขอเว็บเพจธรรมดา โดยปกติจะมีนามสกุลของเอกสารเป็น .htm หรือ .html นั้น เมื่อใช้เว็บไคลเอ็นท์ เปิดดูหรือเรียกดูเว็บเพจใด เว็บเซิร์ฟเวอร์ก็จะส่งเว็บเพจนั้นกลับมายังเว็บไคลเอ็นท์ จากนั้นเว็บไคลเอ็นท์จะแสดงผลตามคำสั่งภาษา HTML ที่อยู่ในเอกสารนั้นๆ



ภาพที่ 1.3 รูปแบบการติดต่อสื่อสารระหว่างเว็บไคลเอ็นท์และเว็บเซิร์ฟเวอร์เพื่อเรียกเว็บเพจธรรมดา

จะเห็นว่าเว็บเพจ จากภาพที่ 1.3 เป็นเว็บเพจแบบคงที่ กล่าวคือผู้ใช้จะพบกับเว็บเพจหน้าตาเดิมๆ ทุกครั้ง จนกว่าผู้ดูแลเว็บจะทำการปรับปรุงเว็บเพจนั้นใหม่ ภาษา HTML เป็นภาษาที่ใช้อธิบายหน้าตาของเว็บเพจ (ภาษา HTML จัดเป็นภาษาในกลุ่มที่เรียกว่า page description language) หรือกล่าวอีกนัยหนึ่งคือ ภาษา HTML สามารถกำหนดให้เว็บเพจมีหน้าตาอย่างที่ต้องการได้ ในปัจจุบันมีการพัฒนาโปรแกรมที่ใช้ในการจัดการเว็บเพจ ให้มีความน่าสนใจมากยิ่งขึ้น

1.4.4 กลไกการทำงานเมื่อร้องขอเว็บเพจที่ใช้ภาษา PHP จะมีรูปแบบการประมวลผลและแสดงผลเป็นแบบไดนามิกเว็บเพจ สามารถติดต่อกับผู้ใช้ ฐานข้อมูล คำนวณ และประมวลผลตามเงื่อนไขต่างๆ ได้ ลักษณะการประมวลผล ดังภาพที่ 1.4



ภาพที่ 1.4 การร้องขอการใช้งานจากเว็บไคลเอ็นท์ไปยังเว็บเซิร์ฟเวอร์ที่ได้ติดตั้ง Engine ของภาษา PHP

จากภาพที่ 1.4 เป็นกระบวนการประมวลผลภาษา PHP โดยเริ่มต้นเมื่อเว็บไคลเอ็นท์ร้องขอไฟล์หรือเอกสารที่ใช้ภาษา PHP ในการพัฒนาร่วมกับภาษา HTML มายังเว็บเซิร์ฟเวอร์ เว็บเซิร์ฟเวอร์จะเรียกใช้ Engine ของภาษา PHP หรือตัวแปลคำสั่งของภาษา PHP กรณีที่มีการใช้คำสั่งในการเชื่อมต่อฐานข้อมูลระบบก็จะติดต่อไปยังฐานข้อมูลตามการร้องขอ มีการตรวจสอบสิทธิ์การเข้าถึงฐานข้อมูลและอื่นๆ ตามลำดับขั้นตอนวิธี ลักษณะการแปลคำสั่งของภาษา PHP นั้นจะแปลและเปลี่ยนคำสั่งจากภาษา PHP เป็นภาษา HTML แล้วส่งกลับไปแสดงผลที่เว็บไคลเอ็นท์ที่ร้องขอ

1.5 MySQL

MySQL เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System) ชนิดหนึ่ง โดยใช้ภาษา SQL (Structure Query Language) เป็นฐานในการพัฒนา MySQL ถูกพัฒนาขึ้นมา ประมาณปี ค.ศ.1995 MySQL จัดอยู่ในกลุ่มของโอเพ่นซอร์ส แต่แตกต่างจากโอเพ่นซอร์สโดยทั่วไป เนื่องจากถูกพัฒนาภายใต้บริษัท MySQL AB ในประเทศสวีเดน ผลิตภัณฑ์ของบริษัทฯ มีทั้งแบบที่อนุญาตให้ใช้งานได้ฟรี และแบบที่ใช้ในเชิงธุรกิจ ทีมพัฒนา MySQL ประกอบด้วย Monty Widenius, David Axmark และ Allan Larsson ที่มาของชื่อ MySQL มาจากชื่อลูกสาวของ Monty Widenius ชื่อ My และใช้สัญลักษณ์ปลาโลมาเป็นสัญลักษณ์ของ MySQL





David Axmark

Allan Larsson

Monty Widenius

ภาพที่ 1.5 ทีมพัฒนาฐานข้อมูล MySQL

ที่มา: Declan McCullagh. (2007)., Week of Innovative Regions in Europe. (2007).
และ Helsinki, Finland. (2009).

ปัจจุบันบริษัทซันไมโครซิสเต็มส์ (Sun Microsystems, Inc.) ได้ซื้อกิจการทั้งหมดของบริษัท MySQL AB ดังนั้นผลิตภัณฑ์ทั้งหมดที่ถูกพัฒนาโดยบริษัท MySQL AB จึงเป็นกรรมสิทธิ์ของบริษัทซันไมโครซิสเต็มส์

1.6 ชนิดของข้อมูลที่ MySQL สนับสนุน

1.6.1 ข้อมูลชนิดตัวเลข (Numeric Data types)

1) BIT ใช้สำหรับเก็บข้อมูลชนิดเลขฐานสองที่มีสัญลักษณ์เพียงสองค่า ประกอบด้วย 0 (ศูนย์) กับ 1 (หนึ่ง) บางครั้งอาจหมายถึงค่าของโอกาสทางเลือก เช่น ปิดกับเปิด ไม่ใช่กับใช่ เท็จกับจริง ซ้ายกับขวา เป็นต้น

2) TINYINT ใช้สำหรับจัดเก็บข้อมูลชนิดตัวเลขจำนวนเต็ม ขนาด 8 บิต

3) SMALLINT ใช้สำหรับเก็บข้อมูลชนิดตัวเลขจำนวนเต็ม ขนาด 16 บิต

4) MEDIUMINT ใช้สำหรับเก็บข้อมูลชนิดตัวเลขจำนวนเต็ม ขนาด 24 บิต

5) INT ใช้สำหรับเก็บข้อมูลชนิดตัวเลขจำนวนเต็ม ขนาด 32 บิต

6) BIGINT ใช้สำหรับเก็บข้อมูลชนิดตัวเลขจำนวนเต็ม ขนาด 64 บิต

7) FLOAT ใช้สำหรับเก็บข้อมูลชนิดตัวเลขจำนวนจริง (เลขที่มีเศษทศนิยม) ขนาด 32 บิต

8) DOUBLE ใช้สำหรับเก็บข้อมูลชนิดตัวเลขจำนวนจริง เช่นเดียวกับ FLOAT ขนาด 64 บิต

1.6.2 ข้อมูลชนิดวันที่และเวลา (Date and Time Data types)

1) DATETIME ใช้สำหรับจัดเก็บข้อมูลวันเดือนปีและเวลา ขนาด 64 บิต แสดงผลในรูปแบบ YYYY-MM-DD HH:MM:SS

2) DATE ใช้สำหรับจัดเก็บข้อมูลเฉพาะข้อมูลวันเดือนปี ขนาด 24 บิต แสดงผลในรูปแบบ YYYY-MM-DD

3) TIME ใช้สำหรับจัดเก็บข้อมูลเฉพาะข้อมูลเวลา ขนาด 24 บิต แสดงผลในรูปแบบ HHH:MM:SS โดยที่ HHH จะคืน ค่าจาก -838 ถึง 838

4) TIMESTAMP ใช้สำหรับเก็บข้อมูลชนิดวันที่และเวลา เช่นเดียวกับ DATETIME แต่จะเก็บในรูปแบบของ YYYYMMDDHHMMSS หรือ YYMMDDHHMMSS หรือ YYYYMMDD หรือ YYMMDD แล้วแต่ว่าจะระบุค่าแอดทริบิวต์ TIMESTAMP[(M)] โดยกำหนด M เป็น 14 หรือ 12 หรือ 8

5) YEAR ใช้สำหรับเก็บข้อมูลปีเท่านั้น ขนาด 8 บิต แสดงผลในรูปแบบ YYYY มีค่าตั้งแต่ 1901 - 2155 เท่านั้น

1.6.3 ข้อมูลชนิดข้อความ (String Data Types)

1) CHAR ใช้สำหรับเก็บข้อมูลชนิดข้อความ ขนาดสูงสุดได้ไม่เกิน 255 อักขระ ถ้าอักขระที่ใ้ยาวไม่เท่ากับที่กำหนดไว้ อักขระที่เหลือ จะถูกแทนที่ด้วยช่องว่าง

2) VARCHAR (ย่อมาจาก Variable Character Field หมายถึง กลุ่มข้อมูลตัวอักขระที่ไม่สามารถระบุความยาวได้) ใช้สำหรับเก็บข้อมูลชนิดข้อความ ชนิดข้อมูลประเภท varchar สามารถเก็บข้อมูลตัวอักขระขนาดเท่าใดก็ได้ที่ไม่เกินความยาวที่จำกัดไว้ (เก็บข้อมูลตามที่ได้รับมาจริงเท่านั้น) ขนาดสูงสุดได้ไม่เกิน 255 อักขระ

3) BINARY ใช้สำหรับเก็บข้อมูลชนิดเลขฐานสองที่มีสัญลักษณ์เพียงสองตัว เหมือน BIT ประกอบด้วย 0 (ศูนย์) กับ 1 (หนึ่ง) บางครั้งอาจหมายถึงการที่มีโอกาสเลือกได้เพียง 2 ทาง เช่น ปิดกับเปิด ไม่ใช่กับใช่ เท็จกับจริง ซ้ายกับขวา เป็นต้น

4) VARBINARY ใช้สำหรับเก็บข้อมูลชนิดข้อความ ขนาดสูงสุดไม่เกิน 8,000 อักขระ มีลักษณะการเก็บคล้าย VARCHARA คือ การเก็บข้อมูลตามที่ได้รับมาจริงเท่านั้น

5) TINYTEXT ใช้สำหรับเก็บข้อมูลชนิดข้อความ ขนาดสูงสุดไม่เกิน 255 อักขระ สามารถใช้งานพีเจอร์ FULL TEXT SEARCH ของ MySQL

6) TEXT ใช้สำหรับเก็บข้อมูลชนิดข้อความ เช่นเดียวกับ TINYTEXT ขนาดสูงสุดไม่เกิน 65,535 อักขระ (64 KB) เหมาะสำหรับเก็บข้อมูลจำพวกเนื้อหาที่มีความยาวหรือจำนวนมาก

7) MEDIUMTEXT ใช้สำหรับเก็บข้อมูลชนิดข้อความ เช่นเดียวกับ TINYTEXT ขนาดสูงสุดไม่เกิน 16,777,215 อักขระ (16 MB)

8) LONGTEXT ใช้สำหรับเก็บข้อมูลชนิดข้อความ เช่นเดียวกับ TINYTEXT ขนาดสูงสุดไม่เกิน 4,294,967,295 อักขระ (4 GB)

9) ENUM ใช้สำหรับเก็บเซตของข้อมูลชุดหนึ่งที่มีจำนวนสมาชิกที่กำหนดไว้แน่นอน และทราบค่าทุกตัว มักจะเป็นข้อมูลที่มีลักษณะคงที่ ขนาดสูงสุดไม่เกิน 65,535 อักขระ

10) SET ใช้สำหรับเก็บข้อมูลที่เป็นกลุ่มของข้อมูลที่ยอมให้เลือกได้ 1 ค่าหรือหลายๆ ค่า สามารถกำหนดได้ถึง 64 ค่า

1.6.4 ข้อมูลชนิด BLOB (Binary Large Object)



1) TINYBLOB ใช้สำหรับเก็บข้อมูลชนิดไบนารี ได้แก่ ไฟล์ข้อมูลต่างๆ เช่น ไฟล์รูปภาพ ไฟล์มัลติมีเดีย เป็นต้น คือ ไฟล์อะไรก็ตามที่อัปโหลด (Upload) ผ่านฟอร์มอัปโหลดไฟล์ในภาษา HTML ขนาดสูงสุดไม่เกิน 256 ไบต์

2) BLOB ใช้สำหรับเก็บข้อมูลชนิดไบนารี เช่นเดียวกับ TINYBLOB ขนาดสูงสุดได้ไม่เกิน 64,000 ไบต์

3) EDIUMBLOB ใช้สำหรับเก็บข้อมูลชนิดไบนารี เช่นเดียวกับ TINYBLOB ขนาดสูงสุดได้ไม่เกิน 16 เมกะไบต์ (16 MB)

4) LONGBLOB ใช้สำหรับเก็บข้อมูลชนิดไบนารี เช่นเดียวกับ TINYBLOB ขนาดสูงสุดได้ไม่เกิน 4 กิกะไบต์ (4 GB)

สรุป

ภาษา PHP เป็นภาษาประเภทเซิร์ฟเวอร์ไซด์สคริปต์ คือ ประมวลผลและแปลภาษาในฝั่งของเว็บเซิร์ฟเวอร์ แล้วส่งกลับมายังเว็บเบราว์เซอร์เพื่อแสดงผลในฝั่งของไคลเอ็นท์ ภาษา PHP เป็นภาษาสคริปต์ที่นิยมใช้ในปัจจุบันและมีแนวโน้มการใช้งานเพิ่มขึ้นในอนาคต สำหรับการพัฒนาไดนามิกเว็บเพจ เนื่องจากโครงสร้างภาษา PHP มีความยืดหยุ่นในการเขียนสคริปต์ มีฟังก์ชันการทำงานที่หลากหลาย สามารถประยุกต์ใช้งานได้ง่ายสำหรับการติดต่อกับผู้ใช้งานระบบ การประมวลผลตามเงื่อนไข การติดต่อกับระบบไฟล์และไดเรกทอรี และมีความสามารถในการเชื่อมต่อกับระบบฐานข้อมูลที่หลากหลาย แต่ที่ได้รับความนิยมมากที่สุด คือ ฐานข้อมูล MySQL

คำถามท้ายบท

1. จงอธิบายหลักการทำงานของภาษาสคริปต์ประเภทไคลเอ็นต์ไซด์สคริปต์
2. จงอธิบายหลักการทำงานของภาษาสคริปต์ประเภทเซิร์ฟเวอร์ไซด์สคริปต์
3. จงความแตกต่างระหว่างเว็บเพจแบบคงที่ และแบบไดนามิก
4. จงอธิบายหลักการทำงานของภาษา PHP เมื่อเว็บไคลเอ็นต์ร้องขอเว็บเพจที่พัฒนาด้วยภาษา PHP มายังเว็บเซิร์ฟเวอร์โดยสังเขป
5. จงอธิบายชนิดของข้อมูลที่ MySQL สนับสนุน พร้อมยกตัวอย่างประกอบ